UNIVERSIDAD POLITÉCNICA DE MADRID

ESCUELA TÉCNICA SUPERIOR
DE INGENIEROS DE TELECOMUNICACIÓN

PROYECTO FIN DE CARRERA

DESARROLLO Y EVALUACIÓN DE UN SISTEMA DE TELEOPERACIÓN DE ROBOTS BASADO EN RECONOCIMIENTO DE GESTOS USANDO EL SENSOR KINECT

ALBERTO GONZÁLEZ DE DIOS

2014

Proyecto Fin de Carrera

Título:	DESARROLLO Y EVALUACIÓN DE UN SISTEMA DE TELEOPERACIÓN DE ROBOTS BASADO EN RECONOCIMIENTO DE GESTOS USANDO EL SENSOR KINECT
Autor:	D. ALBERTO GONZÁLEZ DE DIOS
Tutor:	D. JUAN MANUEL MONTERO MARTÍNEZ
Departamento:	DEPARTAMENTO DE INGENIERÍA ELECTRÓNICA

Miembros del Tribunal

Presidente:	D. JAVIER FERREIROS LÓPEZ
Vocal:	D. JUAN MANUEL MONTERO MARTÍNEZ
Secretario:	D. ROBERTO BARRA CHICOTE
Suplente:	D. RUBEN SAN SEGUNDO HERNÁNDEZ

Los miembros del tribunal arriba nombrados acuerdan otorgar la calificación de:

Madrid, a de de 2014

Resumen del Proyecto

En este Proyecto Fin de Carrera, se presenta un sistema de reconocimiento de gestos para teleoperar robots basado en el sensor Kinect. El proyecto se divide en dos partes, la primera relativa al diseño y evaluación de un sistema de reconocimiento de gestos basado en el sensor Kinect; y la segunda, relativa a la teleoperación de robots usando el sistema de reconocimiento de gestos desarrollado.

En la primera parte, se enumeran las características y limitaciones del sensor Kinect. Posteriormente, se analiza la detección de movimiento y se presenta la máquina de estados propuesta para detectar el movimiento de un gesto. A continuación, se explican los posibles preprocesados de un esqueleto en 3 dimensiones para mejorar la detección de gestos y el algoritmo utilizado para la detección de gestos, el algoritmo de *Alineamiento Temporal Dinámico (DTW)*. Por último, se expone con detalle el software desarrollado de reconocimiento y evaluación de gestos, el *Evaluador de Gestos*, y se realiza un análisis de varias evaluaciones realizadas con distintos perfiles de configuración donde se extraen las conclusiones de acierto, fiabilidad y precisión de cada configuración.

En la segunda parte, se expone el sistema de teleoperación del robots y su integración con el evaluador de gestos: este sistema controla el robot Lego Mindstorm mediante la detección de gestos o el reconocimiento de voz. Por último, se exponen las conclusiones finales del proyecto.

Palabras clave

Reconocedor de gestos, evaluador de gestos, sistema de teleoperación de robots, alineamiento temporal dinámico, dynamic time warping, DTW, entrenamiento de gestos, detección de movimiento de gestos

Agradecimientos

Los agradecimientos siempre tienen un comienzo, un *en primer lugar*, y un orden. Sin embargo, considero injusto establecer un orden a las personas e instituciones que me han ayudado directa o indirectamente a llegar al final del camino que acaba con este proyecto: convertirme en un ingeniero de telecomunicación. Por eso, dividiré los agradecimientos en 4 grupos: mi familia, mi tutor, mis amigos y las instituciones.

Mi Familia

A mi **padre** y a mi **madre**, que tanto lucharon e insistieron para que mis hermanos y yo tuviéramos la mejor formación posible: mi hermano, **doctor ingeniero** de telecomunicación por la Universidad de Valladolid; mi hermana, **arquitecta** por la Universidad Politécnica de Madrid; y yo, presentando este proyecto para ser ingeniero de telecomunicación por la Universidad Politécnica de Madrid.

A mi **hermana**, **Beatriz**, que siendo 2 años mayor que yo, hemos compartido tanto. Gracias por enseñarme en una noche a manejar el 3D Studio Max para poder realizar las animaciones en 3D de este proyecto.

A mi **hermano**, **Óscar**, que aunque 10 años mayor, en parte guió mi camino hacia este ingeniería. Gracias por ayudarme en una tarde a montar el robot Lego Mindstorm para este proyecto.

Mi Tutor

Una mención especial tiene mi tutor, **Juancho**. Nos conocimos hace varios años cuando mi compañero Jorge y yo hicimos una práctica especial de un robot con ultrasonidos, en la asignatura *LCEL*. Posteriormente, realicé con mi

compañero Cristian otra práctica especial, también tutorizada por Juancho, en la asignatura *LSED*: Checkersbot, un robot que jugaba a las damas. Práctica que ganó el diploma de honor a la mejor integración hardware-software.

Instituciones

Me gustaría agradecer en primer lugar al proyecto **INAPRA**, proyecto sin el cuál no se habría este Proyecto Fin de Carrera. Al **Grupo de Tecnología del Habla (GTH)** y al departamento al que pertenece, el **Departamento de Ingeniería Electrónica (DIE)**. Y a esta escuela, **Escuela Técnica Superior de Ingenieros de Telecomunicación (ETSIT)**, y a esta universidad, **Universidad Politécnica de Madrid**, que ha sido mi casa durante muchos años. Gracias a todos por acoger este proyecto y hacerlo realidad.

Amigos

Una mención especial tienen 2 amigos, **Javier de la Rubia** y **Tamara Salmerón**, que desarrollaron un robot clasificador de residuos, Kigo, que incorporaba una parte de detección de gestos y me sirvió de guía en el comienzo del proyecto. ¡Muchas gracias por vuestra inestimable ayuda!

Y por supuesto no me olvido de la asociación **IEEE**, mi verdadera casa durante la carrera y donde tantos amigos he conocido: Jorges, Albertos, José, Javier, Elenas, Pichi, Tamara, Luis, Manu, Willy, Tanis...

Muchas gracias a todos por vuestra ayuda: familia, tutor, amigos e instituciones.

Notas

Todo el material asociado a este proyecto ha sido publicado en la web:

http://alberto.gonzalezdedios.es/pfc

Este material incluye:

- **Anteproyecto**
- **Memoria**
- **Código fuente**
- **Vídeos demostrativos**

El código fuente ha sido publicado en github (https://github.com/albertogd/GestureRecognitionSystem) bajo licencia **MIT** (Expat), para que cualquier persona, departamento, institución o compañía pueda hacer uso de él libremente, con la única condición de que se mencione el origen del software.

El modelo 3D usado para realizar las animaciones, es el modelo número 708030 de la empresa TurboSquid, bajo licencia *Royalty Free License* (http://www.turbosquid.com/3d-models/free-rigged-animators-3d-model/708030).

Esta memoria ha sido realizada en Latex con la plantilla *ezthesis* bajo licencia *MIT* (http://navarroj.com/latex/ezthesis.html).

Software Utilizado

Para la realización de este proyecto, se ha utilizado el siguiente software:

- **MiKTeX 2.9** - Distribución TeX/LaTeX para Windows.
- **TeXnicCenter** - Editor de LaTeX para Windows
- **Visual Studio 2012** - Entorno de desarrollo integrado (IDE) para Windows.
- **MySQL Server 5.6** - Sistema de gestión de bases de datos relacional, multihilo y multiusuario.
- **Adobe Photoshop CS6** - Editor de gráficos rasterizados para el retoque de fotografías y gráficos.
- **Autodesk 3ds Max 2013** - Software de modelado y animación en 3D.
- **Matlab 2013** - Entorno interactivo para desarrollo de algoritmos, visualización de datos, análisis de datos y cálculo numérico.
- **Kinect SDK 1.8** - Kit de desarrollo de software que proporciona las herramientas y APIs, tanto nativas como manejadas, necesarias para desarrollar aplicaciones con Kinect.
- M**icrosoft Speech Platform SDK 11** - Kit de desarrollo de software que incluye los recursos de desarrollo, herramientas y ejemplos para el desarrollo de aplicaciones de voz que utilizan la plataforma de voz de Microsoft en tiempo de ejecución.

Índice general

Índice de figuras

Capítulo 1

Introducción

Los robots móviles teleoperados son herramientas muy utilizadas en diversos campos, desde la industria hasta la medicina. Hay situaciones en las que la intervención del ser humano no es posible, como en la cirugía remota. En otras situaciones, la presencia del una persona supondría un riesgo muy grande para su vida, como en la desactivación de una mina. En todos estos casos, la utilización de robots móviles teleoperados ha permitido poder trabajar en estas situaciones sin la necesidad de la presencia física del ser humano.

La teleoperación de robots siempre se ha realizado a través de un controlador específico para cada tipo de robot. Sin embargo, existen otros dos métodos para teleoperar un robot: mediante la realización de gestos y mediante el reconocimiento de voz. En total, tenemos tres métodos:

1. Manejando un **controlador**: mando, joystick, teclado, ratón...
2. Realizando **gestos**
3. Usando la **voz**

La utilización de un controlador es el sistema más fiable y utilizado hasta la fecha. Sin embargo, suelen ser dispositivos complejos en los que hay que realizar muchos tipos de movimientos con pocos botones. Para cada tipo de dispositivo es necesario aprender combinaciones de botones para realizar cada tipo de acción. Además, estos dispositivos requieren tener las **manos ocupadas**, y los movimientos que transmitimos al robot no son movimientos naturales. Mediante

el reconocimiento de voz o gestos, es posible no sólo manejar robots o sistemas mediante la realización de gestos, sino que además se puede interactuar con estos sistemas de una **manera natural**.

La interacción del ser humano con los sistemas empezó a despertar interés con la aparición de los ordenadores. Las interfaces de usuario evolucionaron de entornos de texto con el uso del teclado, a entornos gráficos con el uso del ratón y finalmente entornos gráficos con pantallas táctiles. Sin embargo estos dispositivos no son convenientes ni naturales para trabajar en Entornos Virtuales.

En el área de las videoconsolas, observamos una evolución más natural que en los ordenadores. Los primeros sistemas de interacción fueron los mandos y joysticks. En 2006 Nintendo introdujo el Wiimote, un controlador que detecta los movimientos realizados con las manos. En 2010 Microsoft lanzó Kinect, un controlador con cámaras y micrófonos que detecta visualmente los movimientos realizados.

(a) Mando arcade

(b) Wiimote

(c) Kinect

Figura 1.1: Evolución de sistemas de interacción

Detrás de estos dispositivos, está la lógica del reconocimiento de los movimientos. A medida que ha evolucionado la tecnología, ha ido aumentando la complejidad de la lógica del reconocimiento de los movimiento:

- Con los mandos y joystick simplemente hay unos botones que hacen de interruptores. Los movimientos se asocian con los botones pulsados.

- El wiimote, aunque es un tipo de mando, tiene una lógica más compleja. Los movimientos se detectan mediante acelerómetros en los 3 ejes y un sensor óptico detecta a donde está apuntando. En función del valor de los acelerómetros se calcula el movimiento.

- El Kinect, que fue lanzado en 2010, es el dispositivo más complejo. Detecta los movimientos a través de reconocimiento visual. Dotado de una cámara rgb y un sensor de profundidad, Kinect reconoce las partes del cuerpo humano a partir de la imagen de profundidad[12].

Este proyecto está centrado en la teleoperación de robots mediante reconocimiento de gestos. Sin embargo, con el objetivo de evaluar las diferencias entre la teleoperación de robots mediante el reconocimiento de gestos y el reconocimiento de voz, se ha añadido también el reconocimiento de voz como medio para teleoperar el robot.

El robot escogido ha sido el robot Lego Mindstorm EV3. El uso de este robot en este proyecto tiene 3 razones fundamentales, gracias a las cuales este proyecto pretende ser un sistema universal que pueda ser probado por cualquier persona en su casa:

- Es muy fácil e intuitivo de montar, por lo que cualquier persona puede implementar este proyecto en su casa
- Es universal y a un precio razonable, que se puede comprar en cualquier tienda de robótica e incluso por internet
- Está diseñado para niños, con lo cual podrán *jugar* con el reconocimiento de gestos hasta los más pequeños.

Muchos han sido los modelos propuestos para el reconocimiento de gestos mediante visión artificial, pero hasta ahora estos modelos habían sido probados con cámaras RGB bajo unas condiciones específicas, siempre favorables al modelo propuesto. Es normal encontrar en diferentes artículos mismos modelos de reconocimiento de gestos con un porcentaje de aciertos totalmente distinto. Estas condiciones van desde utilización de cámaras especiales hasta condiciones de luz específicas o utilización de fondos específicos.

Este proyecto pretende dar un paso más en la literatura, y desarrollar un sistema con reconocimiento de gestos en 3 dimensiones y que pueda ser utilizado por cualquier persona en su propia casa. El sistema de reconocimiento de gestos contribuirá a la difusión y al acercamiento de la tecnología a todo el mundo, y más concretamente del reconocimiento de gestos. Con sólo un Kinect, se podrán hacer evaluaciones de distintos modelos propuestos de reconocimiento de gestos

en multitud de entornos con condiciones ambientales totalmente diferentes.

El uso de Kinect en este proyecto tiene 2 razones fundamentales:

1. Kinect detecta a las personas y envía la posición de 20 puntos del esqueleto en 3 dimensiones. Esto nos evita las tareas de procesamiento de detectar personas y calcular la posición de las partes del esqueleto.

2. Es un sensor que se puede adquirir en cualquier parte del mundo a un precio razonable. Esto permite que este proyecto pueda ser reproducido en cualquier parte del mundo por cualquier persona.

1.1. Objetivos

Este proyecto tiene como propósito *desarrollar y evaluar un sistema de teleoperación de robots basado en reconocimiento de gestos usando el sensor Kinect.* Para lograr este propósito, se han propuesto dos objetivos principales:

- Análisis, diseño y evaluación de un sistema de reconocimiento de gestos usando el sensor Kinect

- Desarrollo de un sistema de teleoperación de robots integrando el reconocedor de gestos realizado

1.2. Fases del Trabajo

La realización de este proyecto se ha divido en dos partes correspondientes a los dos objetivos definidos anteriormente.

1.2.1. Primera parte

En la primera parte, consistente en el *análisis, diseño y evaluación de un sistema de reconocimiento de gestos usando el sensor Kinect*, se han definido las siguientes fases de trabajo:

- Analizar el efecto en el reconocimiento de gestos de la realización de los gestos desde distintas distancias y posiciones y desarrollar un algoritmo

para eliminar los efectos producidos por la posición relativa de la persona al Kinect analizados.

- Analizar la energía del movimiento de los gestos para su detección e implementar una máquina de estados con umbrales de energía y tiempo que permita detectar el inicio y el fin del movimiento de un gesto analizados.
- Adaptar e implementar el algoritmo de alineamiento temporal dinámico (DTW) para alinear secuencias de esqueletos. Definir una medida de distancia entre esqueletos para el alineamiento de gestos.
- Elaborar un sistema de entrenamiento y pruebas para hacer una evaluación por parte de los usuarios y realizar una serie de evaluaciones para obtener datos estadísticos del porcentaje de aciertos
- Grabar vídeos del sistema: realización de un entrenamiento, una evaluación, y la detección de gestos de manera continua.

1.2.2. Segunda parte

En la segunda parte, consistente en el *desarrollo de un sistema de teleoperación de robots integrando el reconocedor de gestos realizado*, se han definido las siguientes fases de trabajo:

- Montar el Lego Mindstorm EV3 con la forma de un humanoide para que pueda repetir los gestos realizados por los usuarios.
- Desarrollar una interfaz que permita comunicar el sistema de reconocimiento de gestos con el robot Lego Mindstorm.
- Analizar e implementar la correspondencia de los gestos reconocidos con los movimientos que tiene que realizar el robot.
- Analizar e implementar un sistema de reconocimiento voz mediante Kinect.
- Desarrollar una interfaz gráfica para teleoperar el robot Lego Mindstorm que se integre con el evaluador de gestos realizado en la primera parte
- Extraer las conclusiones finales del proyecto
- Grabar vídeos del sistema reconociendo gestos y comandos por voz.

Parte I

Análisis, diseño y evaluación de un sistema de reconocimiento de gestos usando el sensor Kinect

Capítulo 2

Estado del Arte

El reconocimiento de gestos es un campo de investigación donde se han propuesto muchos modelos y algoritmos a lo largo de los años. En este capítulo veremos cuatro de los principales modelos de reconocimiento de gestos que se han propuesto en la literatura: **redes neuronales**, **máquinas de estado finitas (FSM)**, **modelos ocultos de Markov (HMM)** y **alineamiento temporal dinámico (DTW)**. Además, introduciremos el concepto de programación dinámica, técnica usada para desarrollar los algoritmos de algunos de los modelos de reconocimientos de gestos que veremos.

2.1. Programación Dinámica

La programación dinámica no es un modelo de reconocimiento de gestos como las Redes Neuronales o los Modelos Ocultos de Markov, sino que es una técnica para implementar algoritmos de una manera más sencilla y eficiente. Se resuelven los problemas identificando una colección de subproblemas abordándolos de uno en uno. Primero se resuelven los más pequeños, usando las respuestas de los más pequeños para calcular o resolver los problemas más grandes, hasta que el problema es resuelto

El algoritmo del cálculo de la distancia más corta usado en el modelo de reconocimiento de gestos del Alineamiento Temporal Dinámico (DTW), y el algoritmo de Viterbi para encontrar la secuencia más parecida de estados ocultos (HMM), están basados en programación dinámica.

2.2. Redes Neuronales

Las redes neuronales[8] [1] son redes de elementos simples interconectadas masivamente en paralelo y con organización jerárquica, las cuales intentan interactuar con los objetos del mundo real del mismo modo que lo hace el sistema nervioso biológico.

2.2.1. Esquema

La red neuronal está constituida por neuronas interconectadas y arregladas en tres capas, aunque puede variar. Los datos entran por medio de la *capa de entrada*, pasan a través de la *capa oculta* y salen por la *capa de salida*.

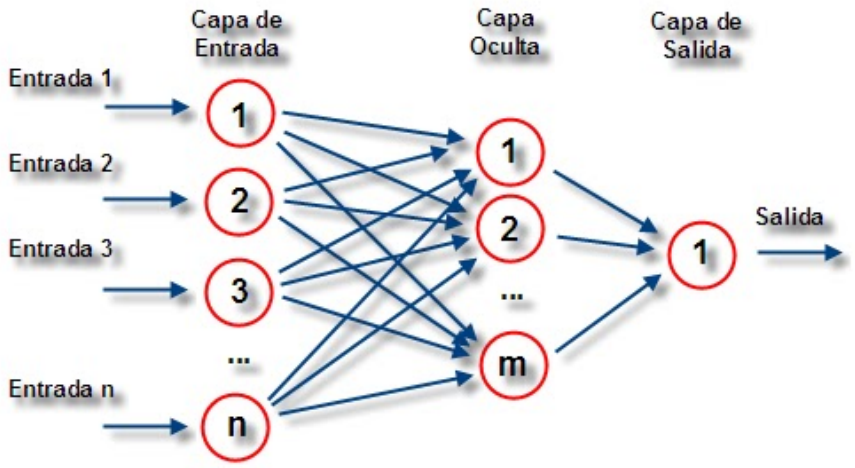


Figura 2.1: Esquema de la red neuronal

Fuente: wikipedia - Licencia GNU GPL

2.2.2. Componentes de las neuronas

Las neuronas tienen 3 componentes: función de entrada, función de activación y función de salida.

Función de Entrada

La neurona tiene una entrada global, ya que toma los valores de entrada como si fueran uno solo. A través de la función de entrada se combinan las entradas:

$$\text{Entrada global} = (entrada_1 \cdot peso_1) * (entrada_2 \cdot peso_2) * \ldots * (entrada_n \cdot peso_n)$$

Los pesos pueden cambiar la medida de influencia que tienen los valores de entrada.

Función de Activación

La función de activación calcula el estado de actividad de una neurona; transformando la entrada global (menos el umbral) en un valor (estado) de activación, cuyo rango normalmente es [0, 1], siendo el 0 totalmente inactiva y 1 totalmente activa.

Función de Salida

El último componente que una neurona necesita es la función de salida. El valor resultante de esta función es la salida de la neurona. La función de salida determina qué valor se transfiere a las neuronas vinculadas. Si la función de activación está por debajo de un umbral determinado, no se le pasa ningún valor a la neurona subsiguiente.

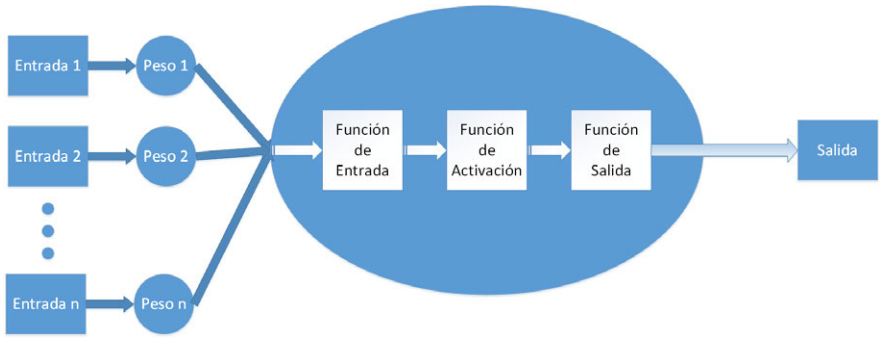


Figura 2.2: Esquema de una neurona

2.2.3. Aprendizaje

Las redes neuronales pueden ser entrenadas para que un valor de entrada produzca un determinado valor de salida. En este contexto, se utilizan un conjunto de pares de valores (entrada,salida) para entrenar una red neuronal.

La identificación de gestos con redes neuronales no es una tecnología tan madura como otros modelos, pero es un campo donde queda mucho margen para la investigación.

2.2.4. Ventajas y desventajas

Debido a su constitución y a sus fundamentos, las redes neuronales artificiales presentan un gran número de características semejantes a las del cerebro. Esto hace que ofrezcan numerosas ventajas y que este tipo de tecnología se esté aplicando en el reconocimiento de gestos. Entre las **ventajas** se incluye:

- Aprendizaje Adaptativo. Los gestos son conjuntos de puntos que siguen unos patrones. Las redes neuronales pueden aprenden a reconocer esos patrones.
- Estos sistemas son generalmente **robustos** y **tolerantes a imprecisiones y ruidos** en los datos de entrada:
    - Tolerancia a fallos: La red puede seguir respondiendo aceptablemente aún si se daña parcialmente.
    - Flexibilidad: La red puede manejar cambios no importantes en la información de entrada, como señales con ruido, sin sufrir cambios drásticos.

Las **desventajas** de usar un modelo basado en redes neuronales son:

- Corregir resultados no es un proceso fácil como puede ser en otros modelos. En las redes neuronales no hay unas reglas que puedas cambiar, sino que hay que buscar las entradas que mejor puedan funcionar en una red neuronal.
- Se necesita un conjunto de datos bastante grande para entrenar la red neuronal. Esto implica la necesidad de dedicar mucho tiempo y recursos en el entrenamiento.

2.2.5. Modelado de la red neuronal como reconocedor de gestos

Modelando el reconocedor de gestos como una red neuronal, las entradas serían los gestos a reconocer, y las salidas las etiquetas asociadas a los gestos.

Para realizar el entrenamiento de la red neuronal, habría que realizar una base de datos con muchos gestos, asociando a cada gesto una etiqueta con la descripción del gesto. En la base de datos tendremos muchas etiquetas iguales asociadas a gestos muy parecidos, pero que no son exactamente iguales. El aprendizaje consistirá en el procesamiento de esta base de datos por parte de la red neuronal.

2.3. Máquinas de Estados Finitas (FSM)

Un gesto puede ser definido como una secuencia ordenada de estados en el tiempo. Con esta definición, se pueden reconocer los gestos modelando el sistema como una máquina de estados finita[5] [6].

La máquina recibe secuencialmente una cadena de símbolos (en el caso del reconocimiento de gestos, puntos en 3 dimensiones de las partes del cuerpo), y cambia de estado por cada símbolo leído o permanece en el mismo estado. Al final de la recepción el estado de la máquina nos indica si la cadena pertenece al Lenguaje que describe nuestra máquina (si reconoce o no el gesto).

2.3.1. Partes de una Máquina de Estados

Las partes que componen una máquina son 5 y se pueden definir:

$$A = \{Q, I, F, \sum, \delta\}$$

donde:

- Q: Conjunto finito de estados
- I: Conjunto de estado iniciales $I \subseteq Q$.

- F: Conjunto de estado finales $F \subseteq Q$.
- $\sum$: Alfabeto finito de entrada.
- δ: Función de Transición $Q \times \sum \rightarrow Q$.

2.3.2. Transición

Supongamos que la máquina se encuentra en el estado q_i donde $q_i \in Q$, también tenemos el símbolo a donde $a \in \sum$. Una entrada a causa que la máquina cambie del estado q_i al estado q_k. La función δ, llamada función de transición, describe este cambio de la forma $\delta(q_i, a) \rightarrow q_k$ de esta forma obtenemos un nuevo estado. Se entiende por transición como el proceso que hace una máquina al cambiar de estado.

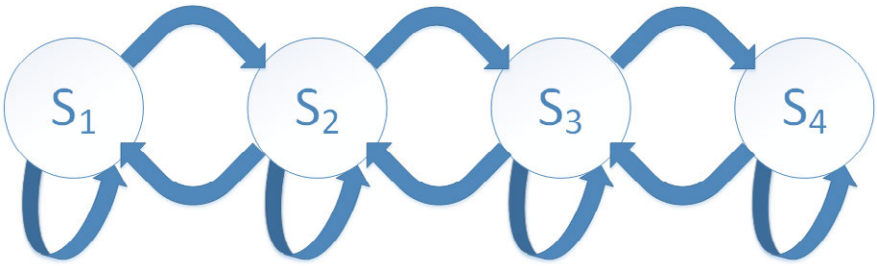


Figura 2.3: Máquina de estados finita de 4 estados

2.3.3. Implementación de una máquina de estados para reconocer de gestos

Para modelar una máquina de estados como un reconocedor de gestos, podríamos tener un estado que fuera subir mano (S_1), y otro bajar mano (S_2). De esta manera podríamos modelar los gestos de la mano que involucren subir o bajar la mano. Una secuencia $S_1, S_1, S_1, S_1, S_1, S_1$ sería un movimiento de levantar la mano durante 6 frames.

2.4. Modelos Ocultos de Markov

Un modelo oculto de Markov[10] (*Hidden Markov Model* en inglés), HMM, es un proceso estocástico que consta de un proceso de Markov no observado (oculto) $Q = \{q_t\}\ \forall\ t\ \in\ \mathbb{N}$ y un proceso observado $O = \{o_t\}\ \forall\ t\ \in\ \mathbb{N}$ cuyas observaciones son estocásticamente dependientes de los estados ocultos.

Estos sistemas evolucionan en el tiempo pasando aleatoriamente de estado a estado y emitiendo en cada momento al azar algún símbolo del alfabeto $\sum$. Cuando se encuentra en el estado $q_{t-1} = i$, tiene la probabilidad a_{ij} de moverse al estado $q_t = j$ en el siguiente instante y la probabilidad $b_j(k)$ de emitir el símbolo $o_t = v_k$ en el tiempo t.

Sólo los símbolos emitidos por el proceso q son observables, pero no la ruta o secuencia de estados q, de ahí el calificativo de öcultode Markov, ya que el proceso de Markov q es no observado.

2.4.1. Procesos de Markov

Sean X, Y, Z variables aleatorias discretas, de tal modo que $X \leftrightarrow Y \leftrightarrow Z$.

Se dice que $X \leftrightarrow Y \leftrightarrow Z$ es una relación markoviana si se cumple que la probabilidad de z condicionada a x e y es igual a la probabilidad de z condicionada a y:

$$p(z\ /\ x, y) = p(z\ /\ y)$$

Un proceso markoviano de orden 1 es un proceso estocástico de orden 1 en el cual su pasado no tiene ninguna influencia en el futuro si su presente está especificado.

En general, un proceso estocástico X_1, X_2, ... se dice que es una cadena de Markov o un proceso Markoviano[3] si $\forall\ n = 1, 2, ...$:

$$Pr(X_{n+1} = x_{n+1} \mid X_n = x_n, X_{n-1} = x_{n-1}, \ldots, X_1 = x_1) = Pr(X_{n+1} = x_{n+1} \mid X_n = x_n)$$

2.4.2. Gestos como modelos de Markov

Los gestos se pueden describir como procesos de Markov donde las posiciones del cuerpo son estados, y los movimientos son transiciones a otro estado. De esta manera podemos modelar cada gesto como un conjunto de transiciones y estados.

2.5. Alineamiento Temporal Dinámico

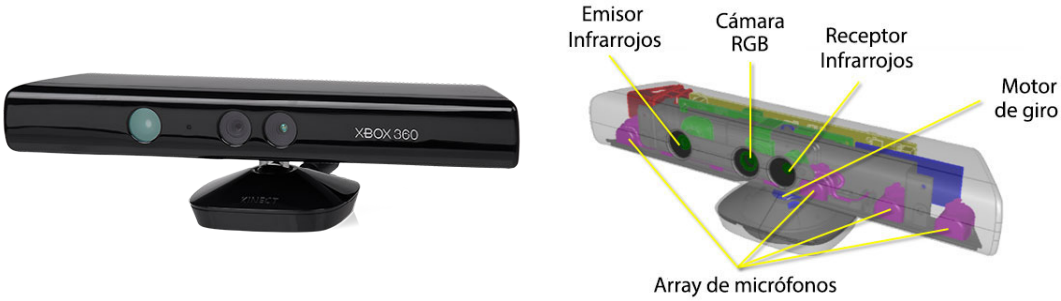
El Alineamiento Temporal Dinámico (DTW) es una técnica o algoritmo para encontrar el alineamiento óptimo entre 2 secuencias temporales. Esta técnica es la utilizad en este proyecto para reconocer gestos, por lo que se explicará con más detalle en el Capítulo 6.

Capítulo 3

Kinect

El sistema de reconocimiento visual de Kinect está formado por una cámara RGB y por un sensor de profundidad (emisor y recepetor de infrarrojos). Kinect envía 3 flujos de información por cada frame capturado: una imagen en color, una imagen de profundidad y los esqueletos detectados. Además, tiene un sistema de giro con un motor integrado que permite inclinar el Kinect y poder ajustar el campo de visión de las cámaras.

El sistema de audio está formado por un array de micrófonos que permite localizar y aislar voces mediante beamforming.



(a) Kinect (b) Elementos de Kinect

Figura 3.1: Kinect

Fuente: Microsoft MSDN (http://msdn.microsoft.com/

3.1. Imagen en Color

Kinect envía directamente la información obtenida por la cámara RGB. Cada frame está compuesto de una matriz de píxeles m x n cuya dimensión viene dada por el parámetro `NUI_IMAGE_RESOLUTION`. Cada píxel contiene el valor de rojo, verde y azul en un punto (x,y) de la imagen en color.

$$\begin{pmatrix} p_{11} & \cdots & p_{1n} \\ \vdots & \ddots & \vdots \\ p_{m1} & \cdots & p_{mn} \end{pmatrix}$$

3.2. Imagen de Profundidad

Kinect envía una imagen en escala de grises obtenida a partir del sensor de profundidad. Cada frame está compuesto de una matriz de píxeles m x n cuya dimensión viene dada por el parámetro `NUI_IMAGE_RESOLUTION`. Cada píxel contiene la distancia en milímetros desde la cámara hasta el objeto más cercano en el punto (x,y) de la imagen.

$$\begin{pmatrix} d_{11} & \cdots & d_{1n} \\ \vdots & \ddots & \vdots \\ d_{m1} & \cdots & d_{mn} \end{pmatrix}$$

Figura 3.2: Imagen en escala de grises.

Fuente: http://fivedots.coe.psu.ac.th/~ad/kinect/ch021/

3.3. Funcionamiento del sensor de profundidad

El sensor de profundidad del Kinect es un escáner 3D de luz estructurada (*Structured-light 3D scanner*). Este tipos de dispositivos miden las distancias de objetos en 3 dimensiones mediante la emisión de patrones de luz y un sistema de cámaras.

Kinect proyecta con el emisor de luz infrarroja un patrón aleatorio de puntos, y la cámara infrarroja captura la luz reflejada. Como el patrón emitido es conocido por Kinect, la cámara puede obtener la información de profundidad a través de algoritmos de visión estéreo, como la correlación cruzada[4] o la triangulación estérea. El tipo concreto de algoritmo usado por Kinect es una información que PrimeSense, el fabricante del sensor de profundidad, no ha desvelado actualmente.

Figura 3.3: Patrón de infrarrojos.

Fuente: https://graphics.stanford.edu/~mdfisher/Kinect.html

El tipo de tecnología usada es necesario para conocer las limitaciones del Kinect. Debido a la emisión del patrón de luz infrarroja, hay que tener en cuenta dos restricciones importantes:

- Kinect sólo se puede utilizar en sitios cerrados. En sitios abiertos la luz solar borraría el patrón.

- No se pueden usar dos Kinects en el mismo sitio, ya que se producirían interferencias entre ambos patrones[13]

3.4. Sistema de Referencia

Kinect procesa la imagen de profundidad y obtiene la información de los esqueletos. La información de los esqueletos contiene puntos en 3 dimensiones de hasta 2 esqueletos humanos detectados en frente del Kinect. La posición del esqueleto y cada uno de sus articulaciones están guardadas como coordenadas (x,y,z)

Al contrario que la imagen de profundidad, las coordenadas del esqueleto están expresadas en metros. El sistema de coordenadas se establece con el Kinect en el origen de coordenadas. El eje X se extiende a lo largo del Kinect, con las X positivas a la izquierda del Kinect. El eje Y se sitúa hacia arriba del Kinect. Y el eje Z, se extiende a lo ancho del Kinect, con las Z positivas hacia la dirección en que apunta el Kinect.

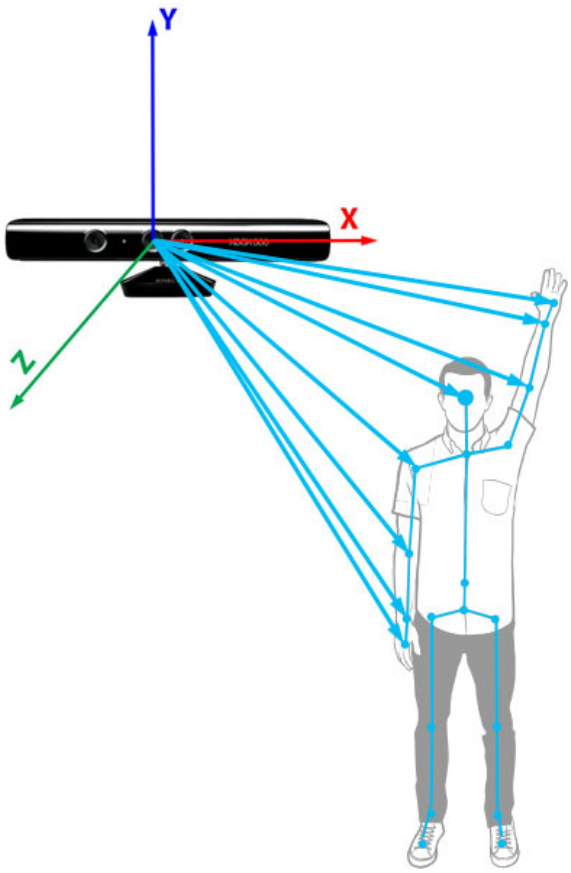


Figura 3.4: Sistema de Referencia de Kinect

3.5. Esqueletos detectados

Kinect ha modelado el esqueleto humano como un conjunto de 20 puntos:

- Tobillo izquierdo
- Tobillo derecho
- Codo izquierdo
- Codo derecho
- Pie izquierdo
- Pie derecho
- Mano izquierda
- Mano derecha
- Cabeza
- Centro de la cadera
- Cadera izquierda
- Cadera derecha
- Rodilla izquierda
- Rodilla derecha
- Centro de los hombros
- Hombro izquierdo
- Hombro derecho
- Columna vertebral
- Muñeca izquierda
- Muñeca derecha

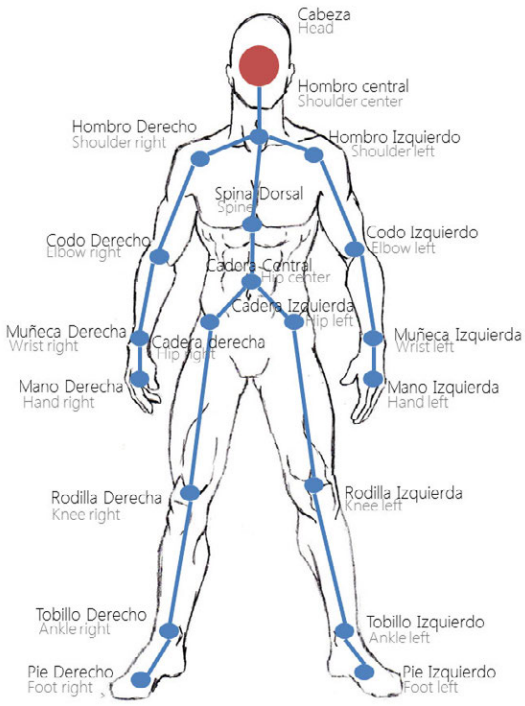


Figura 3.5: Esqueleto

Fuente: http://goodnameblog.blogspot.com.es

Los esqueletos son detectados por Kinect utilizando *decision forests* [11], una combinación de árboles predictores, a través de los cuales estima la posición de los 20 puntos del cuerpo en una imagen dada.

3.6. Campo de Visión

El rango de detección de profundidad de Kinect físico está entre 80 cm y 4 metros, aunque a efectos prácticos se considera que está entre 1'2 y 3'5 metros.

El ángulo de apertura horizontal es de 58º, y el ángulo de apertura vertical es de 43'5º.

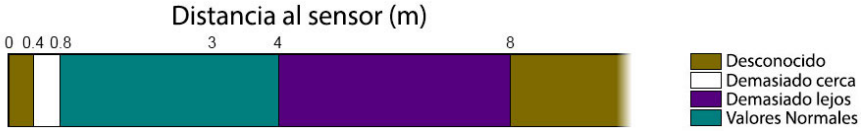


Figura 3.6: Rango del sensor de profundidad

Fuente: Microsoft MSDN (http://msdn.microsoft.com/)

Los límites para las componentes X, Y, Z son:

- X : Rango desde $-3'5 \times \sin \frac{58}{2} = -1'7$ m hasta 1'7 metros.
- Y : Rango desde $-3'5 \times \tan \frac{43'5}{2} = -1'4$ m hasta 1'4 metros.
- Z : Rango desde 1'2 hasta 3'5 metros.

Los rangos de X e Y son en el extremo más alejado, para Z = 3'5 metros. Sin embargo, normalmente operamos en rangos más cercanos, alrededor de Z = 2'5 metros. Para esta profundidad, el rango es:

- X : Desde **- 1'2 metros** hasta **1'2 metros**. Distancia horizontal del Kinect.
- Y : Desde **-1 metro** hasta **1 metro**. Distancia vertical desde el Kinect.
- Z = **2'5 metros**

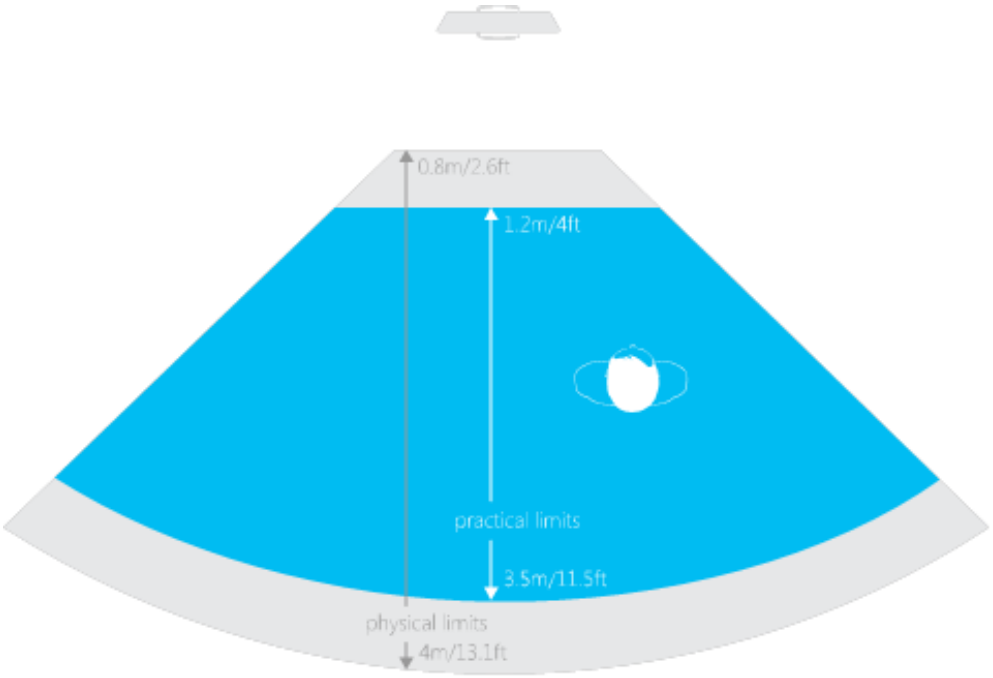


Figura 3.7: Campo de Visión Horiziontal.

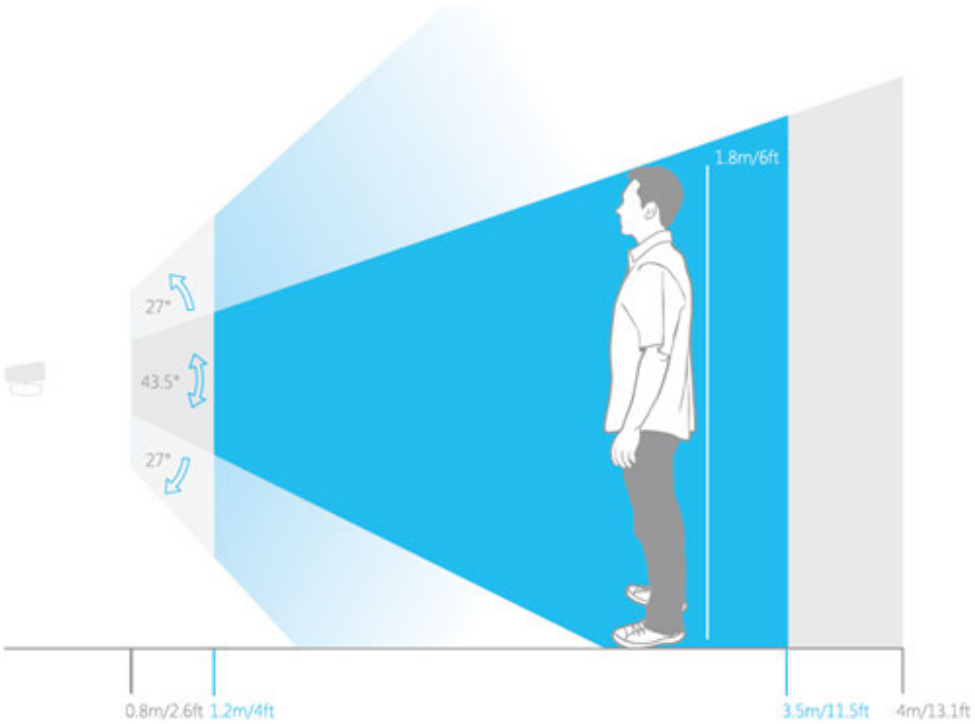


Figura 3.8: Campo de Visión Vertical.

Fuente: Microsoft MSDN (http://msdn.microsoft.com/)

Capítulo 4

Detección del Movimiento

Cuando se analiza un gesto, es necesario saber en qué momento comienza y en qué momento termina el gesto. Para determinar estos instantes, hay que analizar la energía de movimiento en cada instante.

4.1. Energía del Movimiento

La energía del movimiento de un cuerpo, o energía cinética, es la energía que posee ese cuerpo a consecuencia del movimiento. Matemáticamente se define como:

$$E_c = \frac{1}{2}\ m\ v^2$$

En el sistema de coordenadas que utiliza Kinect, sistema de coordenadas cartesianas, la expresión matemática de la energía cinética se transforma en:

$$E_c = \frac{1}{2}\ m\ (\dot{x}^2 + \dot{y}^2 + \dot{z}^2)$$

$$\dot{x} = \frac{dx}{dt},\ \dot{y} = \frac{dy}{dt},\ \dot{z} = \frac{dz}{dt}$$

Los diferenciales de x,y,z respecto al tiempo, $\frac{dx}{dt}$, $\frac{dy}{dt}$, $\frac{dz}{dt}$, se aproximan a partir de la posición de todas las partes del esqueleto enviados por Kinect en cada frame. El diferencial de x respecto a t de una parte del esqueleto, $\frac{dx_i}{dt}$, se aproxima por la diferencia de la posición de la componente x de esa parte del esqueleto en el frame actual y en el frame anterior, dividido entre el tiempo entre

frames. El tiempo entre frames resulta de la inversa de la tasa de frames por segundo (30 FPS):

$$\begin{cases} E_{c_x} = \frac{1}{30} \left(\frac{x_2 - x_1}{t_{frame}}\right)^2 \\ E_{c_y} = \frac{1}{30} \left(\frac{y_2 - y_1}{t_{frame}}\right)^2 \\ E_{c_z} = \frac{1}{30} \left(\frac{z_2 - z_1}{t_{frame}}\right)^2 \\ t_{frame} = \frac{1}{30} = 33,33ms \end{cases}$$

$$E_c = \frac{1}{2}\ m \left[\left(\frac{x_2 - x_1}{0{,}033}\right)^2 + \left(\frac{y_2 - y_1}{0{,}033}\right)^2 + \left(\frac{z_2 - z_1}{0{,}033}\right)^2 \right]$$

Considerando que todas las partes del cuerpo contribuyen de la misma manera al movimiento, es decir, que todas tienen la misma masa, y normalizando la ecuación por la constante $k = \frac{2 \cdot (0{,}033)^2}{m}$, obtenemos la ecuación de la E**nergía cinética normalizada** para los movimientos realizados:

$$\boxed{E_c = (x_2 - x_1)^2 + (y_2 - y_1)^2 + (z_2 - z_1)^2}$$

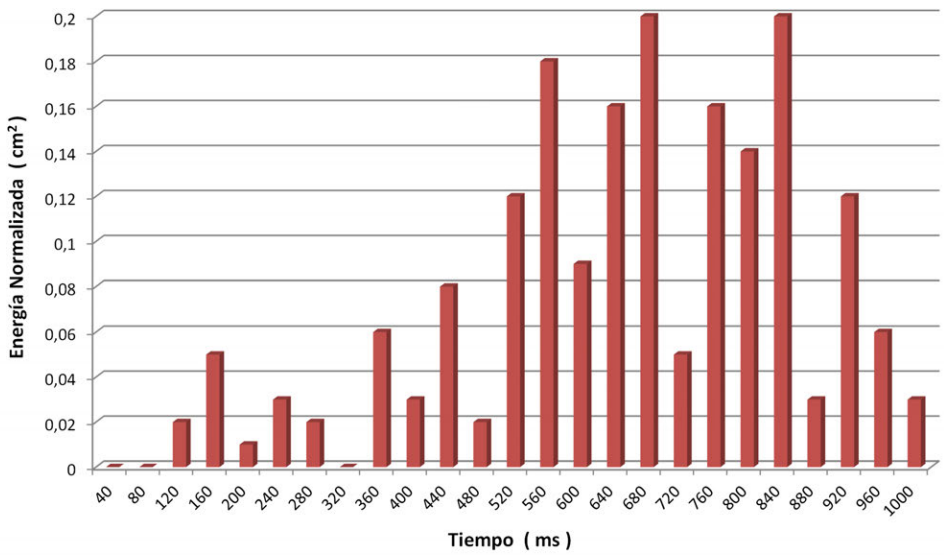


Figura 4.1: Grafica de **Energía Cinética Normalizada** de un Movimiento

4.2. Movimiento de un Gesto

Un gesto se define como el movimiento del rostro, de las manos o de otras partes del cuerpo con que se expresan diversos afectos del ánimo.

Para detectar un gesto es necesario hallar los instantes en que se inicia y se termina el movimiento del gesto. Sabiendo estos instantes, obtendremos el gesto como la secuencia de todas las posiciones del esqueleto desde el instante inicial hasta el instante final hallados. Una vez obtenido el movimiento completo del gesto, podrá ser procesado mediante una técnica de reconocimiento de gestos.

4.2.1. Ruido

El principal problema que se plantea en la detección del movimiento de un gesto es el ruido. El ruido se genera principalmente por 2 factores: un factor de naturaleza humana, y otro factor de naturaleza computacional.

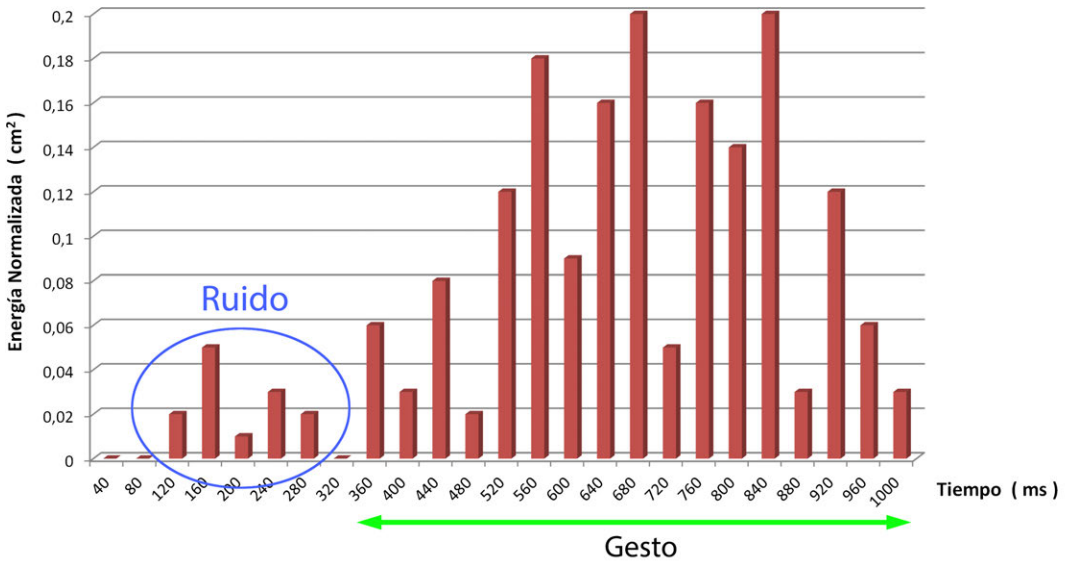


Figura 4.2: Ruido en un gesto

El factor de naturaleza humana es debido a que por naturaleza humana es prácticamente imposible que una persona se quede totalmente quieta sin variar la posición un centímentro.

El factor de naturaleza computacional es debido a que el reconocimiento de las partes del esqueleto humano realizado por Kinect no es un proceso perfecto, sino que que se basa en la estimación. Un punto del esqueleto puede varias de posición varios centímetros aunque la persona no se haya movido. Como se aprecia en la siguiente imagen, la detección de hombros y codos tienen un cierto margen de error.

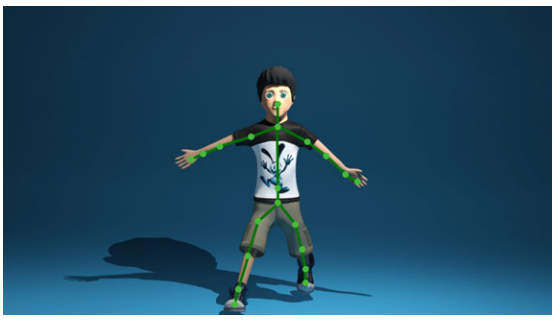


Figura 4.3: Estimación de esqueleto humano por Kinect

4.2.2. Umbral de Energía

Si se considera el comienzo de un gesto cuando la energía del movimiento es mayor que cero, se detectarán gestos que en realidad son sólo ruido. En cuanto se produzca un poco de ruido, se interpretará como el inicio de un gesto.

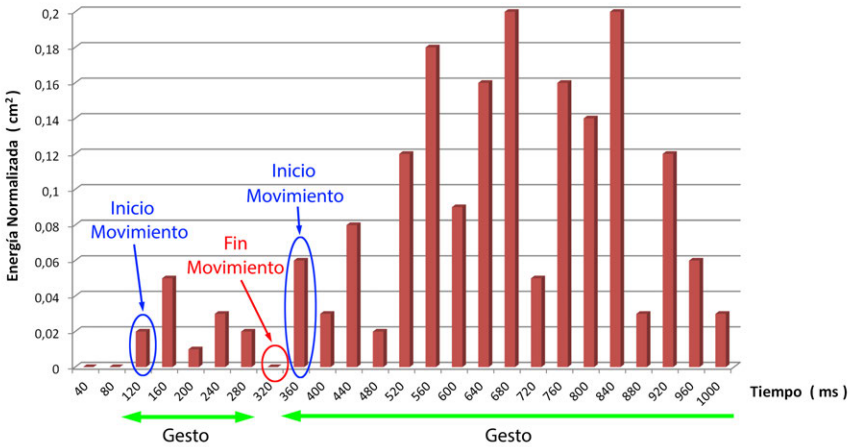


Figura 4.4: Deteccion de gestos cuando la Energía es mayor que 0

Con el objetivo de filtrar el ruido, se introduce el concepto de umbral de energía u_E. Se define el umbral de energía como la energía mínima necesaria para

cambiar de estado. Considerando 2 estados básicos, reposo y gesto, superar el umbral supone iniciar el gesto; y caer por debajo del umbral, terminar el gesto.

Al establecer un umbral de energía bajo, $u_E = 0,02$, la detección sigue siendo vulnerable al ruido. Las oscilaciones del ruido hacen que se detecten falsos inicios y finales de gestos.

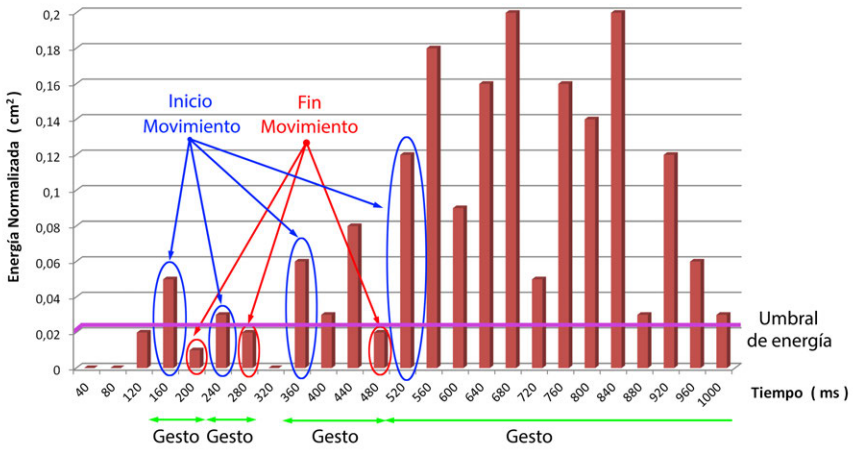


Figura 4.5: Deteccion de gestos con umbral de energía Ue = 0,02

Un umbral de energía bajo no es capaz de filtrar correctamente el ruido, y detecta falsos gestos.

Al establecer un umbral de energía más alto, $u_E = 0,08$, ya no se detectan falsos gestos en la zona de ruido, pero ahora se producen falsos finales de gesto durante la realización del gesto.

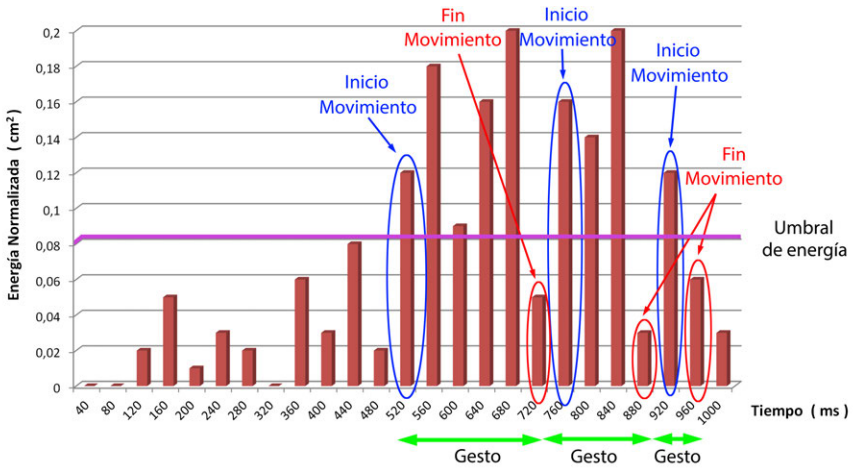


Figura 4.6: Deteccion de gestos con umbral de energía Ue = 0,08

Un umbral de energía alto es conveniente para detectar el inicio de un gesto, ya que filtra muy bien el ruido. Sin embargo, tiene el gran inconveniente de que también filtra parte del comienzo del gesto.

El umbral de energía es una herramienta bastante útil en la filtración del ruido, pero por sí sola no basta para detectar correctamente un gesto. Es necesario analizar si la energía supera el umbral sólo un instante debido al ruido, o si supera el umbral durante más tiempo debido a que es un gesto.

4.2.3. Umbral de Tiempo

Se define el umbral de tiempo como la duración mínima de tiempo necesaria durante la cual la energía de un movimiento tiene que estar superando un cierto umbral para cambiar de estado. Existen dos tipos de umbrales de tiempo: umbral de tiempo continuo y umbral de tiempo acumulado.

Umbral de Tiempo Continuo

Se define el umbral de tiempo continuo como la duración mínima de tiempo durante la cual un movimiento tiene que superar un cierto umbral de energía de forma continua.

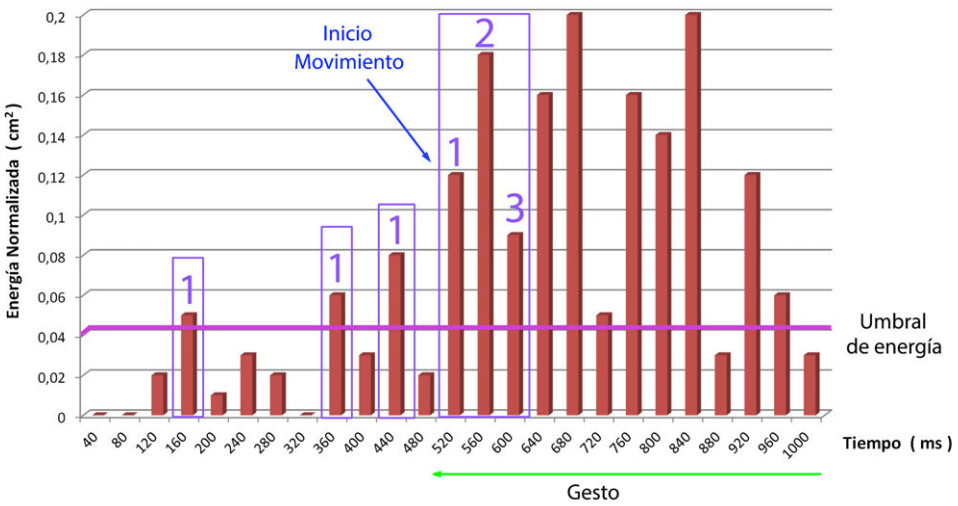


Figura 4.7: Umbral de Tiempo Continuo

En este ejemplo se ha establecido un umbral de tiempo continuo, $u_T = 120$ ms = 3 frames. Hasta que no hay 3 frames consecutivos con valores por encima del umbral de energía $u_E = 0,04$ (línea de color morado), no se considera que supera el umbral de tiempo continuo.

Umbral de Tiempo Acumulado

Se define el umbral de tiempo acumulado como la duración mínima de tiempo durante la cual un movimiento tiene que superar un cierto umbral de energía. Esta cantidad de tiempo se computará como la suma de tiempos parciales que un movimiento sobrepasa un umbral de energía.

En el siguiente ejemplo se establece un umbral de tiempo acumulado, u_T = 120 ms = 3 frames. Cuando se acumulan 3 frames con valores por encima del segundo umbral de energía $u_E = 0,04$ (línea de color morado), se considera que supera el umbral de tiempo acumulado.

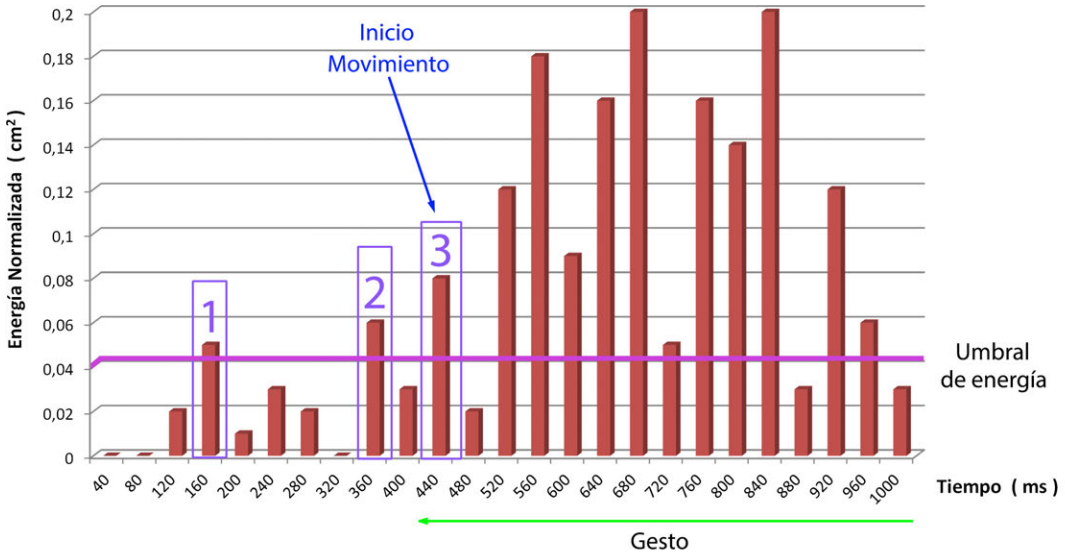


Figura 4.8: Umbral de Tiempo Acumulado

Con el umbral de tiempo el sistema es más fuerte frente al ruido, ya que ahora solo se considera un gesto si supera el umbral de energía durante un intervalo de tiempo. En los ejemplos de tiempo continuo y tiempo acumulado, no se detecta ningún gesto falso en la zona de ruido. Sin embargo, el comienzo del gesto (t = 360 ms) no se detecta correctamente.

4.2.4. Máquina de Estados Genérica

Para filtrar el ruido es necesario aplicar umbrales de energía y tiempo. Sin embargo, aún con estas herramientas, el movimiento de un gesto no se termina de reconocer perfectamente. Un umbral de energía demasiado bajo provoca que el ruido no se filtre, y se detecten falsos movimientos de gestos con el ruido. Un umbral demasiado alto filtra y elimina el problema del ruido, pero elimina también el inicio de los gestos.

Modelando una máquina de estados, con varios umbrales de energía y tiempo, se consiguen unos umbrales dinámicos en función del estado en que se encuentre el movimiento. Para la detección de gestos se propone una máquina de 4 estados: **Reposo**, **Inicio de Movimiento**, **Movimiento** y **Fin de Movimiento**.

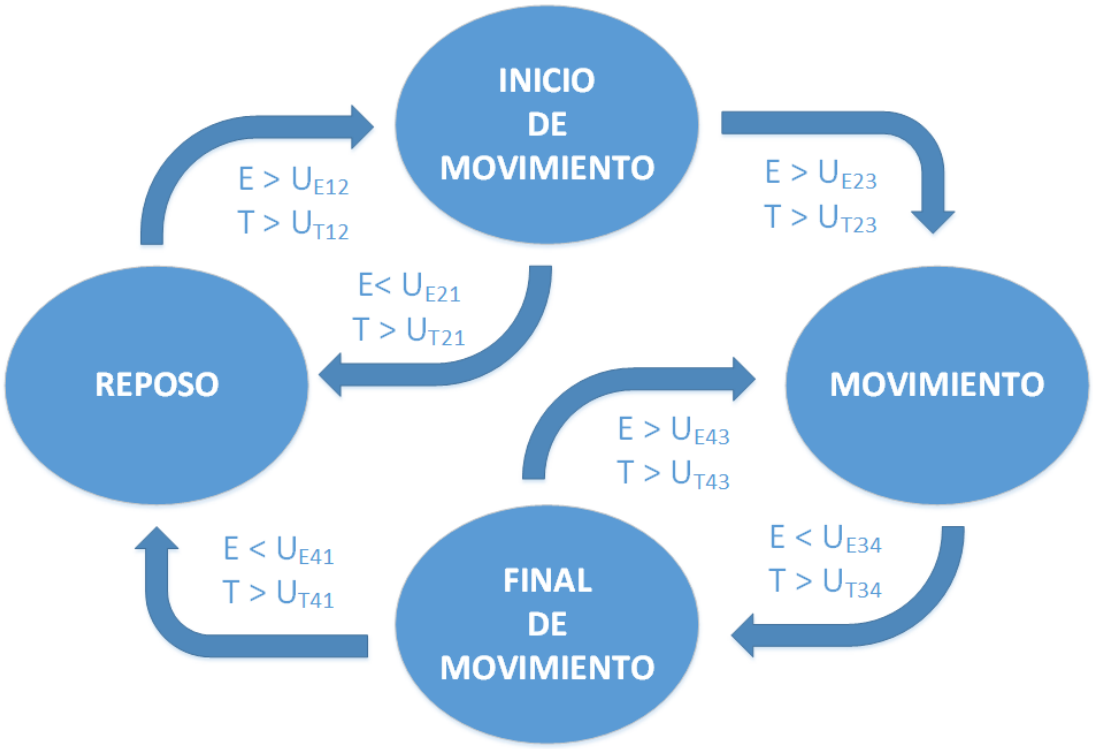


Figura 4.9: Máquina de Estado Genérica

Estados

A la hora de detectar el movimiento de un gesto, se definen 4 posibles estados: reposo, inicio de movimiento, movimiento y final de movimiento.

Reposo Estado inicial. Estado en el que no hay movimiento, sólo lo que se considere seguro que es ruido. Matemáticamente se definde como Reposo $\Leftrightarrow E < u_{E_1}$.

Si se supera el primer umbral de energía ($E > u_{E_1}$), se considera que puede ser un movimiento de un gesto y se pasa al estado de Inicio de Movimiento.

Inicio de Movimiento Estado en el que hay movimiento pero no se sabe si es causado por el ruido o porque se trata de un gesto. Matemática se define como Inicio de Movimiento $\Leftarrow u_{E_1} \leq E < u_{E_2}$.

Se puede llegar a este estado desde el estado de Reposo: si en el estado de Reposo se detecta movimiento ($E > u_{E_1}$), pero no se sabe si es ruido o es un movimiento de un gesto, se pasa al estado de Inicio de Movimiento.

Si se supera el segundo umbral de energía durante un tiempo acumulado $T_1 : E > u_{E2} \forall \Delta t_{ac} > u_{T1}$, se considera que el movimiento es un gesto y se pasa al estado de Movimiento.

Si la energía cae por debajo del primer umbral ($E < u_{E1}$), se considera que es ruido y se regresa al estado de Reposo.

Movimiento Estado en el que se está realizando un gesto. Matemáticamente se define como Movimiento $\Leftarrow E \geq u_{E_2}$

Se puede llegar a este estado desde el estado Inicio de Movimiento si se detecta que efectivamente hay movimiento, o desde el estado final del movimiento si se detecta que no se ha terminado el movimiento.

Si la energía del movimiento cae por debajo del segundo umbral de energía ($E < u_{E_2}$) durante un tiempo acumulado T_2, se considera que el movimiento se está terminando y se pasa al estado Final del Movimiento.

Final del Movimiento Estado en el que se está llegando al final del gesto. Final del Movimiento $\Leftarrow u_{E_1} \leq E < u_{E_2}$

Se llega a este estado cuando el movimiento está llegando a su fin, pero no es seguro si realmente ha finalizado o si ha sido una falsa alarma y el movimiento del gesto continúa.

Si se supera el segundo umbral de energía ($E > u_{E_2}$) se considera que el gesto sigue realizándose y se regresa al estado de Movimiento.

Si la energía cae por debajo del primer umbral ($E < u_{E_1}$), se considera que efectivamente el gesto ha terminado y se pasa al estado de Reposo.

Funcionamiento

Cuando se realiza un gesto, se empieza en un estado de Reposo. Si la energía del movimiento supera un primer umbral de energía, se considera que es un posible movimiento, por lo que se pasa al estado Inicio de Movimiento. Con el fin de no perder los valores del movimiento si resulta que efectivamente es un gesto, se guardan los valores del movimiento. Si ese posible movimiento detectado cae por debajo del primer umbral de movimiento, se considera que era ruido, se descartan los valores guardados y se vuelve al estado inicial de reposo. Pero si el posible movimiento supera el segundo umbral, se considera que efectivamente es un gesto y se pasa al estado de Movimiento. Una vez en el estado de Movimiento, si la energía del movimiento cae por debajo del segundo umbral, se considera que se está terminando el gesto, y se pasa al estado de Final del Movimiento. Si la energía vuelve a superar el segundo umbral, se vuelve al estado de Movimiento. Pero si la energía cae por debajo del primer umbral de energía, se considera que el movimiento del gesto se ha terminado. Se guardan los valores del movimiento almacenados desde el inicio del movimiento hasta este momento, y se pasa al estado de Reposo.

4.2.5. Máquina de Estados por defecto

La máquina de estados genérica permite tener una gran flexibilidad en los umbrales de energía y tiempo. En cada estado se puede configurar qué umbral de energía y tiempo es necesario para cambiar de estado. Sin embargo, tienen un gran inconveniente, y es que este sistema de detección de gestos tiene 12 variables. Con el fin de tener una máquina de estados con valores por defecto, se ha propuesto un caso particular de la máquina de estados genérica donde:

$$\begin{cases} U_{E_{12}} = U_{E_1} \\ U_{E_{21}} = U_{E_1} \\ U_{E_{23}} = U_{E_2} \\ U_{E_{34}} = U_{E_2} \\ U_{E_{43}} = U_{E_2} \\ U_{E_{41}} = U_{E_1} \end{cases} \qquad \begin{cases} U_{T_{12}} = 0 \\ U_{T_{21}} = 0 \\ U_{T_{23}} = U_{T_1} \\ U_{T_{34}} = 0 \\ U_{T_{43}} = U_{T_2} \\ U_{T_{41}} = 0 \end{cases}$$

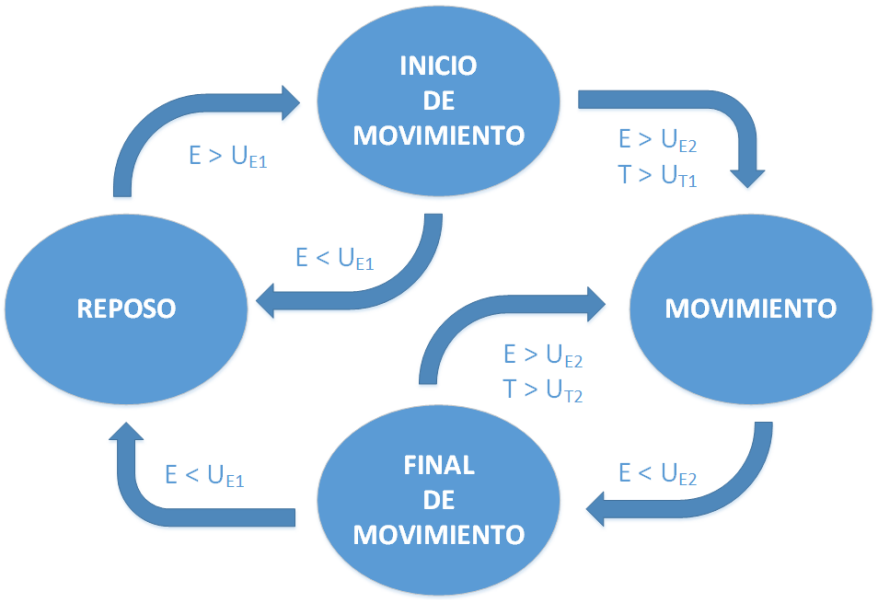


Figura 4.10: Máquina de Estado Propuesta

Umbrales

La máquina de estados propuesta tiene 4 umbrales: 2 umbrales de energía y 2 umbrales de tiempo acumulado.

Primer Umbral de Energía u_{E_1} Todo movimiento cuya energía quede por debajo de este umbral ($E < u_{E_1}$), será considerado ruido. Los movimientos que queden por encima del umbral ($E > u_{E_1}$), es posible movimiento.

Se establecerá este umbral con un valor bajo con el doble compromiso de filtrar ruido pero sin que pueda corresponder a un posible movimiento.

Segundo Umbral de Energía u_{E_2} Todo movimiento cuya energía supere este umbral durante un cierto tiempo, es considerado un gesto. Lo que quede por debajo de este umbral es posible movimiento ($u_{E_1} < E < u_{E_2}$) o ruido ($E < u_{E_1} < u_{E_2}$) .

Se establecerá este umbral con un valor intermedio/alto, superior al primer umbral de energía: $u_{E_2} > u_{E_1}$.

Umbral de tiempo acumulado de Inicio de Movimiento u_{T_1} Tiempo mínimo que un movimiento tiene que superar el segundo umbral de energía, $E > u_{E_2}\ \forall\ \Delta t_{ac} > u_{T_1}$, para que sea considerado un gesto.

Umbral de tiempo acumulado de Fin de Movimiento u_{T_2} Tiempo mínimo que un movimiento tiene que estar por debajo del segundo umbral de energía, $E < u_{E_2}\ \forall\ \Delta t_{ac} > u_{T_2}$, para que se considere que un gesto está en el estado Final del Movimiento.

Ejemplo

Aplicando la máquina de estados propuesta con umbrales de energía $u_{E_1} = 0,01$, $u_{E_2} = 0,04$ y umbrales de tiempo $u_{T_1} = u_{T_2} = 3 frames$, se detecta perfectamente el gesto, con la diferencia de que el estado Inicio de Movimiento es más largo cuando se usan umbrales de **tiempo continuo**.

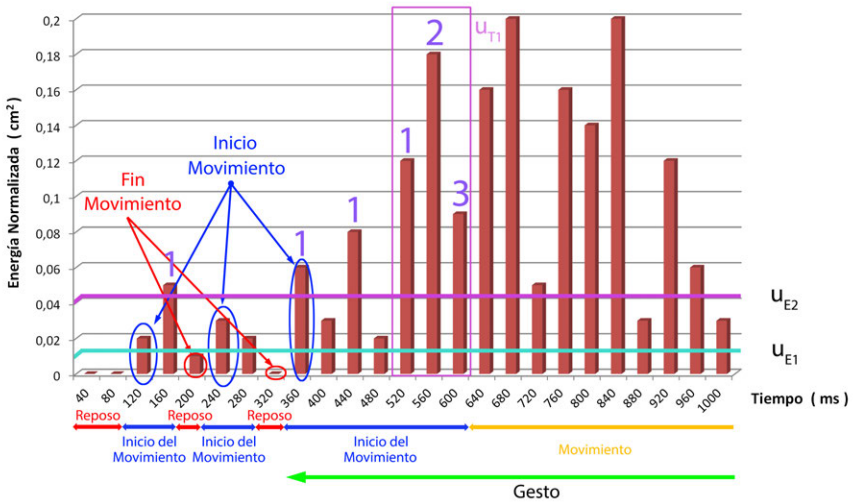


Figura 4.11: Detección de gestos con maquina de estados (tiempo continuo)

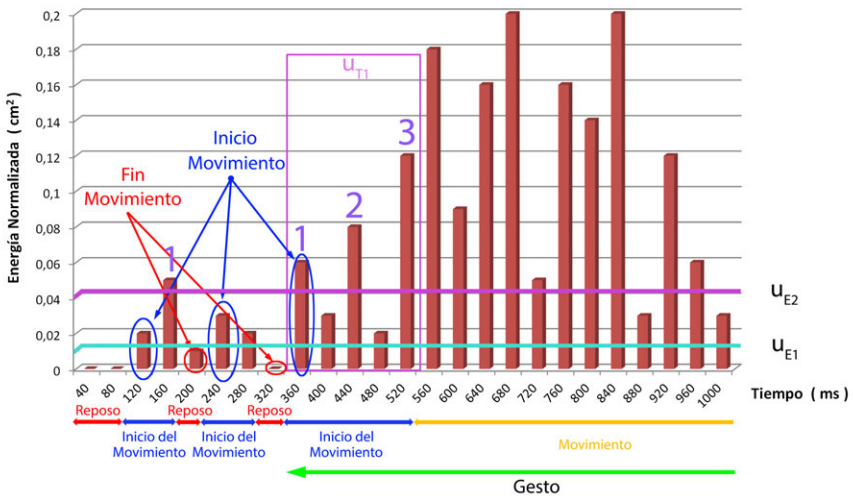


Figura 4.12: Detección de gestos con maquina de estados (tiempo acumulado)

Capítulo 5

Preprocesamiento del Esqueleto

Uno de los principales problemas cuando comparamos dos gestos, aunque la persona haga exactamente el mismo gesto, es la posición relativa de la persona que realiza el gesto respecto al sensor Kinect.

En una situación ideal, la persona estaría siempre situada de frente y a la misma distancia del sensor Kinect. Sin embargo, en una situación real, la persona no está situada siempre totalmente de frente al Kinect ni a la misma distancia. En los mejores casos, la posición variará unos centímetros y el ángulo respecto al Kinect será de unos grados; y en los peores casos, la posición variará varios metros y el ángulo puede superar los 45º.

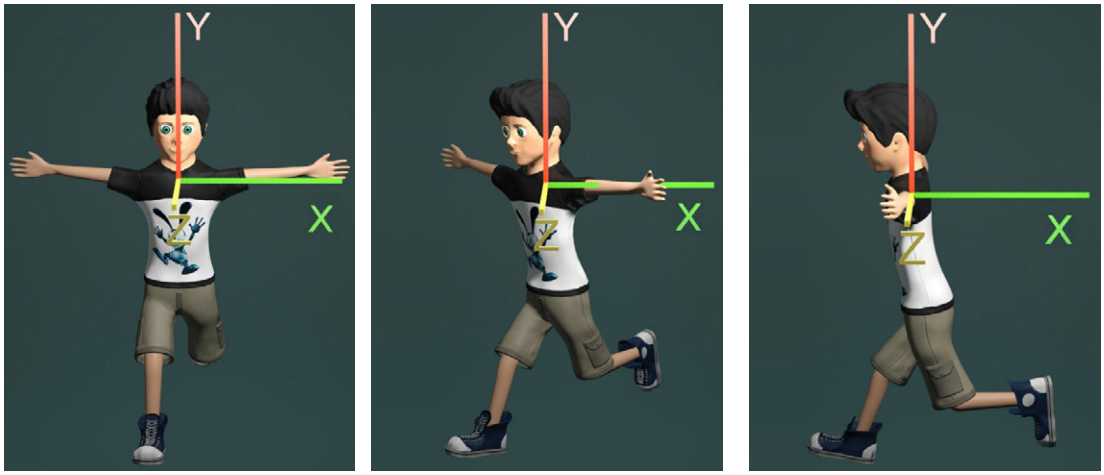
En la siguiente secuencia, se muestra un gesto consistente en el desplazamiento de la mano izquierda de arriba a abajo.

(a) Arriba (b) Transición (c) Abajo

Figura 5.1: Gesto Mano Izquierda de Arriba a Abajo

Si se compara siempre este gesto, el movimiento en el espacio siempre será una curva en el plano XY. Sin embargo, si se realiza ese mismo gesto pero desde otra posición relativa al Kinect, el movimiento ya no es una curva en el plano XY, sino que dependiendo del giro realizado por la persona, el movimiento será una curva en un plano distinto.

En la siguiente secuencia, vemos distintas posiciones relativas a la hora de realizar un gestos: realizado de frente, girado 45º respecto al Kinect y girado 90º respecto al Kinet.



(a) De Frente (b) Girado 45º (c) Girado 90º

Figura 5.2: Gesto realizado en distintas posiciones

Representando cada movimiento en un plano y en una gráfica en 3 dimensiones, observamos las diferentes curvas descritas por el mismo gesto: levantar la mano izquierda.

- En el primer movimiento, la mano describe una curva en el plano XY.
- En el segundo movimiento, girado 45º, la mano describe una curva en el plano X=Z.
- En el tercer movimiento, girado 90º, la mano describe una curva en el plano YZ.

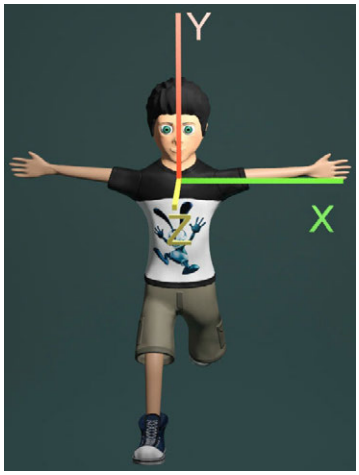


Figura 5.3: Movimiento mano izquierda sin girar (Arriba)

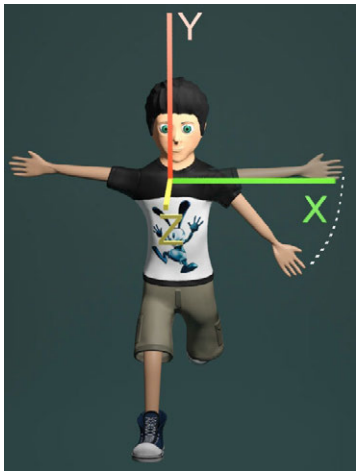


Figura 5.4: Movimiento mano izquierda sin girar (Transición)

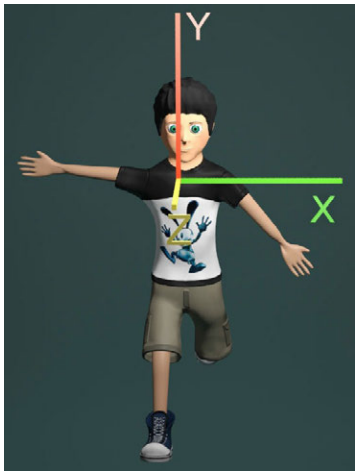


Figura 5.5: Movimiento mano izquierda sin girar (Abajo)

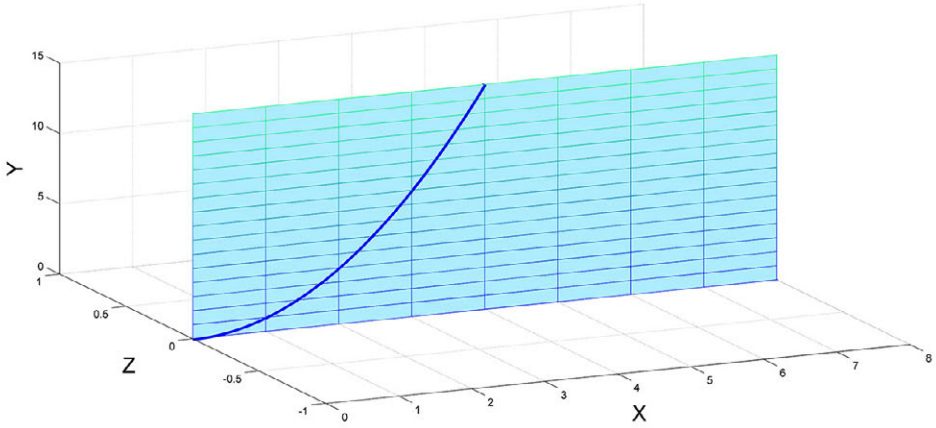


Figura 5.6: Curva del movimiento sin girar (Plano XY)

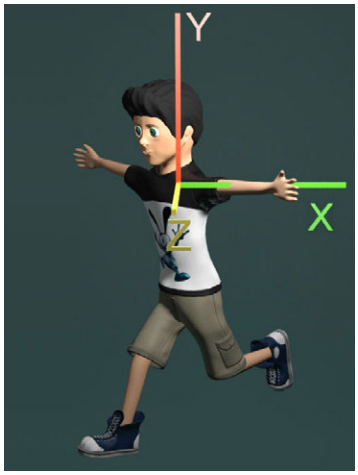


Figura 5.7: Movimiento mano izquierda girado 45º (Arriba)

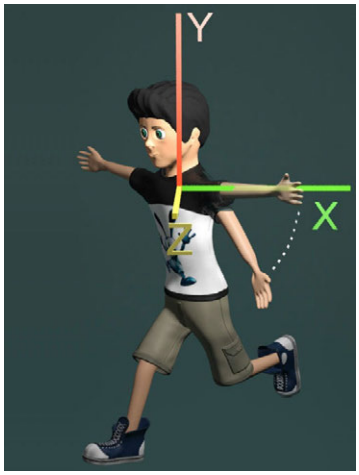


Figura 5.8: Movimiento mano izquierda girado 45º (Transición)

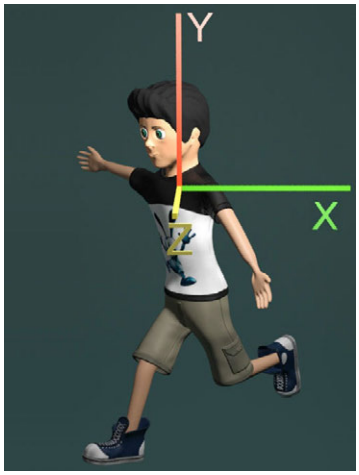


Figura 5.9: Movimiento mano izquierda girado 45º (Abajo)

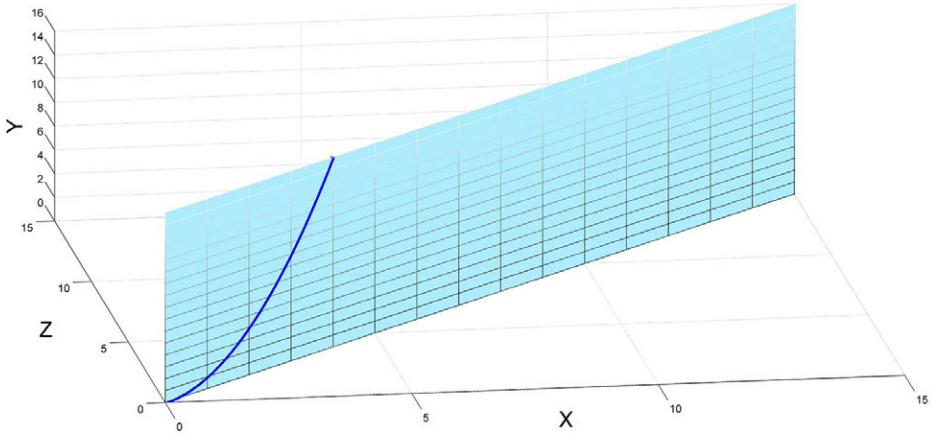


Figura 5.10: Curva del movimiento girado 45º (Plano X=Z)

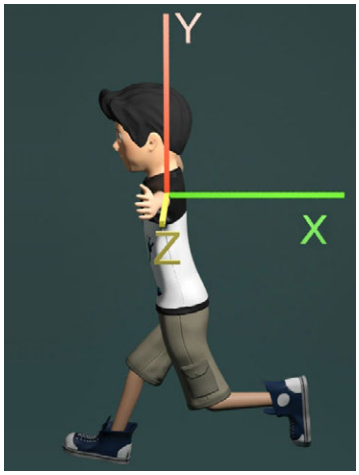


Figura 5.11: Movimiento mano izquierda girado 90º (Arriba)

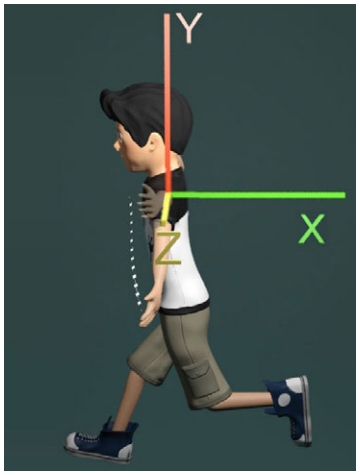


Figura 5.12: Movimiento mano izquierda girado 90º (Transición)

Figura 5.13: Movimiento mano izquierda girado 90º (Abajo)

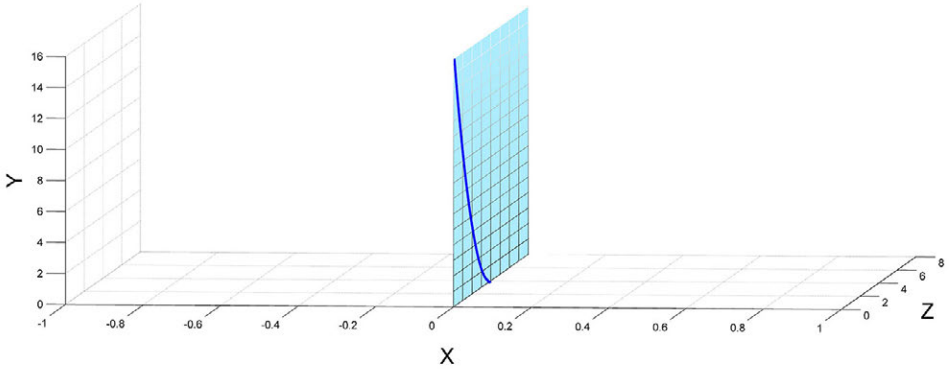


Figura 5.14: Curva del movimiento girado 90º (Plano YZ)

Al realizar el mismo movimiento a distancias distintas del Kinect, la distancia descrita por el movimiento varía. Si la persona está más cerca del Kinect, la curva que describe el recorrido de los movimientos será más larga. En cambio cuando la persona está más lejos, la curva será más corta.

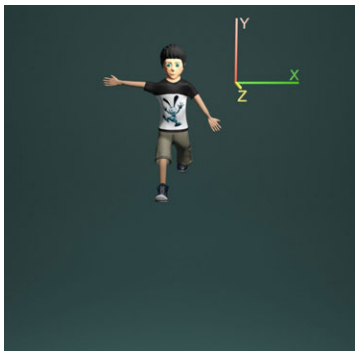


Figura 5.15: Movimiento mano izquierda lejos

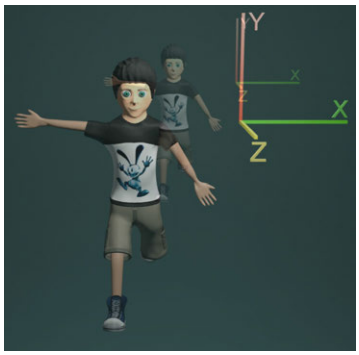


Figura 5.16: Movimiento mano (Transición)

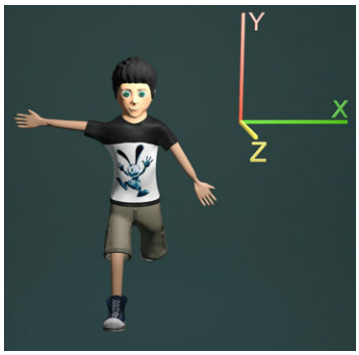


Figura 5.17: Movimiento mano izquierda más cerca

Debido a los efectos que introduce la posición relativa de la persona que realiza el gesto respecto al sensor Kinect, se realizará un preprocesamiento de las coordenadas para que no estas no varíen independientemente de la posición relativa respecto al sensor.

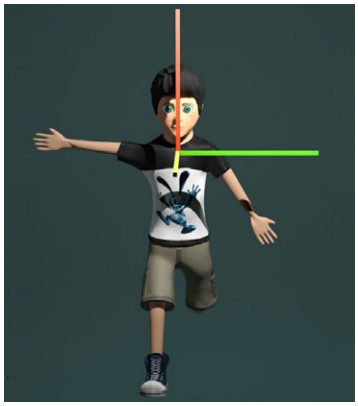


Figura 5.18: Movimiento mano izquierda

Figura 5.19: Movimiento mano izquierda girado 45º

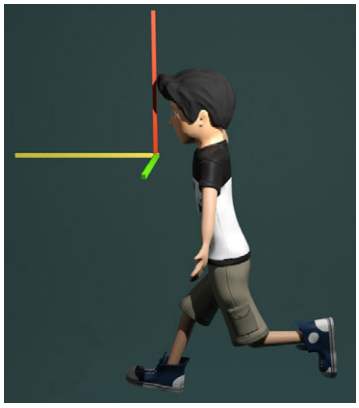


Figura 5.20: Movimiento mano izquierda girado 90º

En la figura se puede apreciar que ahora los movimientos son los mismos independientemente de la posición relativa al Kinect. Todos los movimientos estarán descritos en el plano XY formado por las líneas roja y verde, siendo la línea amarilla la normal al plano.

5.1. Transformaciones Lineales en 3 Dimensiones

Los objetos visualizados por Kinect quedan ubicados en el espacio en su Sistema Universal de Referencia (SUR), donde el eje X se define como la línea recta a lo largo del Kinect, el eje Z se define como la línea recta a lo ancho del Kinect, y el eje Y la perpendicular al plano formado por los ejes X y Z.

Todas las coordenadas de los objetos, incluidas las coordenadas de los puntos del esqueleto, estarán en este sistema de coordenadas.

Cuando la persona cambia su posición con respecto al sistema de referencia (Kinect), se está realizando una transformación líneal. En el caso de que la persona gire, se está realizando una rotación. Si se aleja o se acerca, un cambio de escala. Y si se mueve, si está produciendo una traslación.

5.1.1. Coordenadas homogéneas

Sea (x,y,z) las coordenadas cartesianas de un punto P, se definen las coordenadas homogéneas de ese punto P como (x,y,z,h) donde h es un número real no nulo.

Un sistema de coordenadas homogéneo es el sistema resultante de añadir una dimensión a un sistema de coordenadas dado. Cuando se convierte un punto en coordenadas cartesianas (x, y,z) a una representación homogénea (x_h, y_h, z_h, h), las ecuaciones que contienen x, y, z se convierten en ecuaciones homogéneas de cuatro parámetros x_h, y_h, z_h, h.

Expresar las coordenadas cartesianas como coordenadas homogéneas permite representar todas las ecuaciones de transformación geométrica como multiplicaciones de matrices.

5.1.2. Traslación

La traslación de un punto $P = (x, y, z, 1)$ una distancia $d = (d_x, d_y, d_z)$ queda determinada por la multiplicación de P por la matriz de traslación T:

$$\begin{cases} P' = (x', y', z', 1) \\ x' = x + dx \\ y' = y + dy \\ z' = z + dz \end{cases}$$

$$P' = (x', y', z', 1) = (x, y, z, 1) \cdot T$$

$$T = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ d_x & d_y & d_z & 1 \end{pmatrix}$$

La traslación se realiza para que el centro de coordenadas sea el centro de la cadera. Sea $c = (c_x, x_y, c_z)$ el punto que representa el centro de la cadera, la matriz de traslación es:

$$T = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ c_x & c_y & c_z & 1 \end{pmatrix}$$

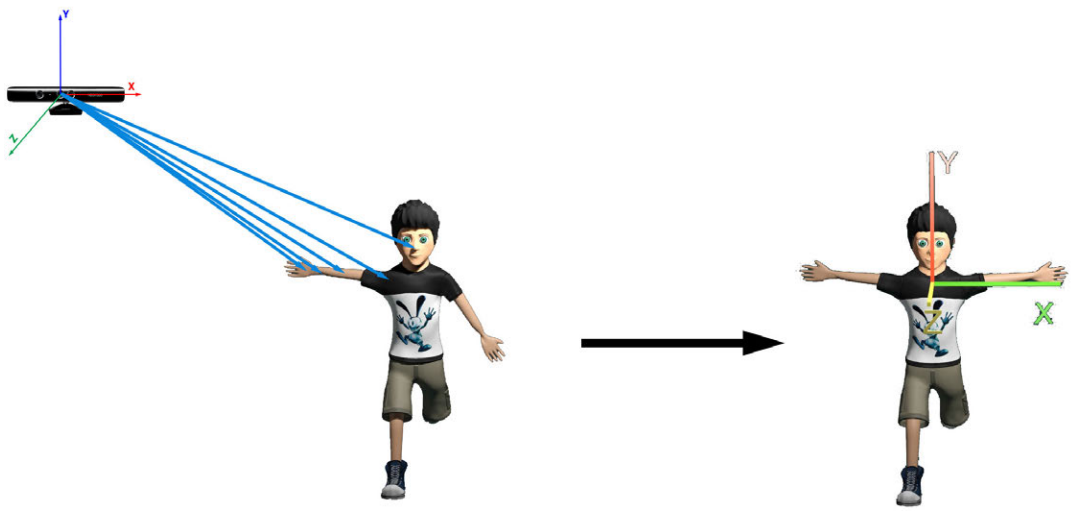


Figura 5.21: Traslación

5.1.3. Escalación

El cambio de escala de un punto $P = (x, y, z, 1)$ en una escala (S_X, S_Y, S_Z) queda determinado por el producto del punto P por la matriz de escalado S:

$$\begin{cases} P' = (x', y', z', 1) \\ x' = x \cdot S_X \\ y' = y \cdot S_Y \\ z' = z \cdot S_Z \end{cases}$$

$$P' = (x', y', z', 1) = (x, y, z, 1) \cdot S$$

$$S = \begin{pmatrix} S_X & 0 & 0 & 0 \\ 0 & S_Y & 0 & 0 \\ 0 & 0 & S_Z & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

La escalación se hará en los ejes X, Y, Z por un factor, como puede ser la distancia entre hombros. De esta manera el esqueleto estará siempre normalizado.

Sea $a = (x_a, y_a, z_a)$ el punto que representa el hombro derecho, $b = (x_b, y_b, z_b)$ el punto que representa el hombro izquierdo:

$$d = \sqrt{(x_a - x_b)^2 + (y_a - y_b)^2 + (z_a - z_b)^2}$$

$$S = \begin{pmatrix} \frac{1}{d} & 0 & 0 & 0 \\ 0 & \frac{1}{d} & 0 & 0 \\ 0 & 0 & \frac{1}{d} & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

5.1.4. Rotación

En 3 dimensiones, para rotar un objeto un ángulo θ es necesario establecer un eje de rotación. Cuando rotamos un objeto en 2 dimensiones realmente estamos utilizando implícitamente como eje de rotación la perpendicular a los ejes de coordenadas.

El proceso de rotación[9] consiste en 3 pasos:

1. Se traslada la persona de forma que el eje de giro del objeto quede alineado con el eje global. El eje de giro es la columna vertebral de la persona; y el eje global, el eje Y. Siendo $v = (v_x, v_y, v_z)$ un punto de la columna vertebral, la matriz de traslación para que la columna vertebral de la persona quede alineada con el eje Y es:

$$T = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ -v_x & 0 & -v_z & 1 \end{pmatrix}$$

2. Se gira el objeto sobre el eje X un ángulo θ, utilizando la matriz de giro:

$$T = \begin{pmatrix} \cos(\theta) & 0 & \sin(\theta) & 0 \\ 0 & 1 & 0 & 0 \\ -\sin(\theta) & 0 & \cos(\theta) & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

3. Se traslada el objeto a su posición original, utilizando la matriz de traslación inversa a la del paso 1):

$$T = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ v_x & 0 & v_z & 1 \end{pmatrix}$$

5.2. Preprocesamiento del Esqueleto

Al reconocer un esqueleto es normal que la posición respecto a la que se ha grabado en el entrenamiento no sea la misma. Para que al comparar el gesto reconocido y el de entrenamiento estén en la misma posición, debemos centrar el esqueleto en un mismo punto. En lugar de realizar un centrado con una matriz con coordenadas homogéneas que implican multiplicaciones de matrices, se ha desarrollado un algoritmo mucho más eficiente consistente en la suma de un vector de traslación.

Kinect proporciona las coordenadas reales de los puntos, por lo que no hay problemas de escala introducidos por la perspectiva. Esto no significa que todos los esqueletos tengan el mismo tamaño, ya que al comparar dos esqueletos que están a distinta profundidad nos parecerá más pequeño el que esté más lejano. Pero simplemente con normalizar la coordenada Z en los dos esqueletos tendremos dos esqueletos a la misma escala. Por tanto en el preprocesamiento del esqueleto la transformación de escalación se convierte simplemente en una traslación en el eje Z, proporcionada ya al centrar el esqueleto en un mismo punto.

Supongamos un esqueleto centro en X = 0, que está a 1 metro de distancia, cuya distancia entre hombros es de 40 cm. El hombro derecho estará en (-0'2, 0, 1) y el izquierdo en (0'2, 0, 1). Este mismo esqueleto a una distancia a 2 metros tendrá el hombro derecho en (-0'2, 0, 2) y el izquierdo en (0'2, 0, 2). Visto desde Kinect parecerá más pequeño el segundo esqueleto, pero las distancias son reales, y simplemente es necesario normalizar la profundidad (coordenada Z) para que ambos esqueletos sean iguales.

Para solucionar el problema de comparar un gesto entrenado de frente con otro gesto realizado girado, introducimos en el preprocesamiento del esqueleto la transformación de rotación para que el esqueleto esté siempre de frente. Para ello consideramos que la línea imaginaria que une los hombros debe ser siempre el eje X. La rotación se hará primero hallando el ángulo entre la línea imaginaria que une los hombros y el eje X, y posteriormente rotando todo el esqueleto este ángulo.

5.2.1. Centrado del esqueleto

En el evaluador de gestos se ha implementado un algoritmo para centrar un esqueleto consistente en trasladar el esqueleto a un punto central. En lugar de utilizar coordenadas homogéneas que implican matrices y multiplicaciones, se ha optado por un algoritmo más óptimo usando un vector de traslación. Mediante este algoritmo se consigue centrar el esqueleto situando el centro de la cadera en el punto (0,0,1):

1. **Se obtiene el vector de traslación como la resta del punto central de la cadera y el punto (0,0,1)**:

   $$\vec{v_t} = (0 - c_x,\ 0 - c_y,\ 1 - c_z) = -\,(c_x,\ c_y,\ c_z - 1)$$

2. **Se traslada cada punto: se suma a cada punto el vector de traslación hallado**:

   $$p' = (p_x - c_x,\ p_y - c_y,\ p_z - c_z + 1)$$

El punto (0,0,1) se ha escogido porque es el origen de coordenadas de la X e Y, y en el eje Z una distancia de 1 metro permite ver bien el esqueleto al ser reproducido. El plano Y = 0 no es el suelo, sino el plano paralelo al suelo a la altura del Kinect. En este punto estamos centrando el esqueleto en la misma coordenada X e Y donde está el Kinect, y a 1 metro de distancia (profundidad) del Kinect.

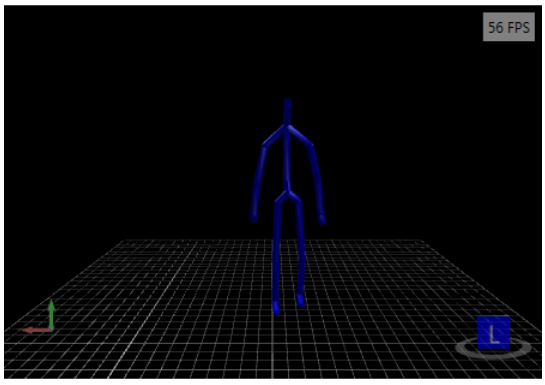


Figura 5.22: Gesto Original

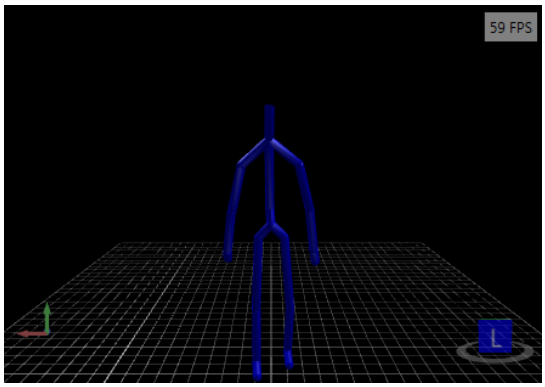


Figura 5.23: Gesto Centrado

5.2.2. Rotación

Rotación mediante Cambio de Base

En un espacio vectorial V, dadas dos bases B y B', la matriz de cambio de base (o de cambio de coordenadas) de B a B', $M_{BB'}$ es la matriz cuyas columnas son las coordenadas de los vectores de la base B expresados en función de la base B'.

Conocidas las coordenadas de un vector en base B, la matriz de cambio de base nos permite hallar las coordenadas de dicho vector en base B'.

Sea $(a_1, a_2, \cdots, a_n)_B \in V$ las coordenadas de un vector en base B, sea $(b_1, b_2, \cdots, b_n)_{B'} \in V$ las coordenadas de ese vector en base B', y sea $M_{BB'}$ la matriz de cambio de base de B a B'. Entonces:

$$M_{BB'} \cdot \begin{pmatrix} a_1 \\ a_2 \\ \vdots \\ a_n \end{pmatrix} = \begin{pmatrix} b_1 \\ b_2 \\ \vdots \\ b_n \end{pmatrix}$$

Siendo la base B, el sistema de referencia de Kinect. Y la base B', el sistema de referencia formado por el esqueleto humano.

Para calcular la matriz de cambio de base $M_{BB'}$, partimos de los vectores de la nueva base expresado en el sistema de referencia de Kinect. Con estos vectores se obtiene la matriz $M_{B'B}$, que es la inversa de $M_{BB'}$.

Algoritmo Para calcular la matriz de cambio de base $M_{BB'}$:

1. Se obtienen los vectores de la base formada por el esqueleto:

   Base del esqueleto $\begin{cases} \textbf{Eje X} - \text{Línea imaginaria que une los hombros} \\ \textbf{Eje Y} - \text{Columna Vertebral (Línea que une la cadera} \\ \qquad\qquad \text{y el punto central entre los hombros)} \\ \textbf{Eje Z} - \text{Recta Perpendicular al eje X e Y} \end{cases}$

   Sean $a = (a_x, a_y, a_z)$ el punto de la cadera, y sea $b = (b_x, b_y, b_z)$ el centro entre los hombros, $c = (c_x, c_y, c_z)$ el hombro derecho e $d = (d_x, d_y, d_z)$ el

hombro izquierdo:

$$\textbf{Base del esqueleto} \begin{cases} \text{Eje X} = U = (c_x - d_x, c_y - d_y, c_z - d_z) \\ \text{Eje Y} = V = (a_x - b_x, a_y - b_y, a_z - b_z) \\ \text{Eje Z} = W = U \times V \end{cases}$$

2. Una vez obtenidos los vectores de la base, se forma la matriz de cambio de base B' a B, $M_{B'B}$, como la matriz cuyas columnas son los vectores de la base obtenidos.

3. Se calcula la matriz de cambio de base B a B', $M_{BB'}$, como la inversa de la matriz $M_{B'B}$

Matriz Inversa La matriz de cambio de base buscada es la inversa de la matriz de cambio de base obtenida en el apartado anterior.

$$\boxed{M_{BB'} = M_{B'B}^{-1}}$$

Para calcular la matriz inversa podemos emplear varias métodos, siendo uno de ellos el método de Gauss-Jordan.

Método de Gauss-Jordan Si tenemos una matriz cuadrada A de orden n, formamos una hipermatriz (A | I) de orden n x 2n, donde en la parte izquierda está la matriz A y en la parte derecha la matriz identidad de orden n. Mediante transformaciones fila (sustitución de una fila por ella misma más una combinación lineal de las restantes) transformamos la hipermatriz original en una que tenga en la parte izquierda la matriz identidad. Entonces, la matriz en la parte derecha será la inversa de la matriz A.

Rotación mediante Giro

El algoritmo de rotación implementado consiste en:

1. **Se obtiene el ángulo que forman la línea imaginaria que une los hombros y el eje X.**

   El ángulo se obtiene calculando el arcocoseno del producto escalar de la línea que une los hombros y el eje X.

2. **Se halla la matriz de rotación.**

   En C# no existe un método directo para obtener la matriz de rotación a partir de un ángulo, pero si se puede hallar de una forma indirecta, y es usando el cuaternión unitario de rotación. C# permite rotar una matriz pasando como parámetro un cuaternión unitario de rotación: rotando la matriz identidad con el cuaternión unitario de giro obtenido, hallaremos la matriz de giro buscada.

   a) Se crea el cuaternión unitario de rotación pasándole como parámetros el eje de giro y el ángulo de giro (en grados) alrededor del eje especificado. El eje de giro es el eje Y, sentido hacia abajo: (0, -1, 0). El ángulo de giro es el hallado en el primer paso, convertido en grados.

   b) Se rota la matriz indentidad usando el cuaternión unitario de rotación obtenido.

3. **Se rota cada vector con la matriz de rotación.**

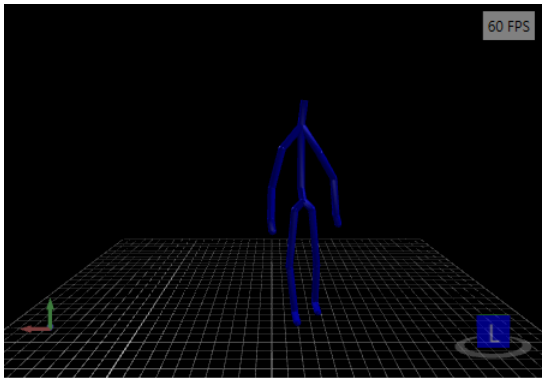


Figura 5.24: Gesto Girado Original

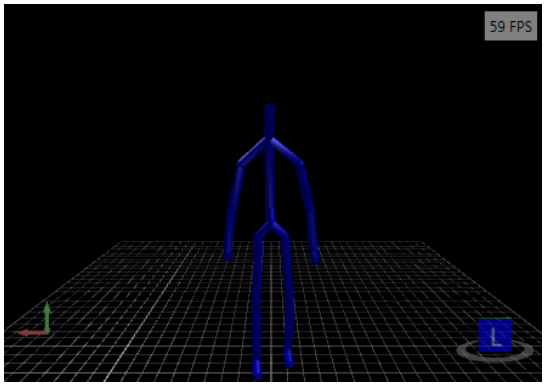


Figura 5.25: Gesto Girado Rotado

Capítulo 6

Alineamiento Temporal Dinámico

El Alineamiento Temporal Dinámico (DTW) es una técnica o algoritmo para encontrar el alineamiento óptimo entre 2 secuencias temporales. El DTW no es una técnica nueva, pero ha sido bastante utilizada desde que se publicara en 1959 Bellman y Kalaba[2]. Utilizada principalmente en aplicaciones de reconocimiento de voz, también se usa en otras áreas, como en medicina para alinear secuencias de proteínas o en música para procesar señales acústicas.

El objetivo de este algoritmo es comparar dos señales discretas en el tiempo $X = (x_1, x_2, ..., x_n)$ de longitud $N \in \mathbb{N}$ e $Y = (y_1, y_2, ..., y_n)$ de longitud $M \in \mathbb{N}$. Aplicado en el reconocimiento de gestos, las secuencias a comparar son esqueletos humanos muestreados cada 33,33 ms (30 fps).

Cada esqueleto $E = (e_1, e_2, ..., e_n)$ es un conjunto de 20 puntos $e_i = (p_1, p_2, ..., p_{20})$ en 3 dimensiones $p_i = (x_i, y_i, z_i)$:

$$\begin{cases} E = (e_1, e_2, ..., e_n) \ \ \forall\ n \ \in \mathbb{N} \\ e_i = (p_1, p_2, ..., p_{20}) \ \ \forall\ i \ \in [1, n] \\ p_i = (x_i, y_i, z_i) \ \ \forall\ j \ \in [1, 20] \end{cases}$$

Para comparar dos secuencias de esqueletos diferentes E1 y E2, es necesario establecer una definición de **distancia entre esqueletos**.

6.1. Distancia entre esqueletos

La distancia entre dos esqueletos E es la aplicación $d : E \times E \Rightarrow \mathbb{R}$ que suma ponderadamente la distancia entre los puntos correspondientes de cada esqueleto.

Por definición de distancia entre esqueletos:

$$d(a,b) = \sum_{i=1}^{20} c_i \cdot d(p_{ia}, p_{ib}) \quad \forall\ a,b \in E$$

Por definición de distancia, verifica las siguientes propiedades:

$$\begin{cases} D1) & d(a,b) > 0 \quad \forall\ a,b \in E \\ D2) & d(a,b) = 0 \ \Rightarrow\ a = b \quad \forall\ a,b \in E \\ D3) & d(a,b) = d(b,a) \quad \forall\ a,b \in E \\ D4) & d(a,c) \leq d(a,b) + d(b,c) \quad \forall\ a,b,c \in E \end{cases}$$

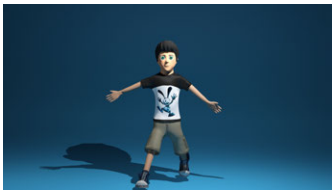
La distancia entre dos puntos del esqueleto p_a, p_b se define como la distancia euclídea entre esos dos puntos:

$$d^2(p_a, p_b) = (p_{xa} - p_{xb})^2 + (p_{ya} - p_{yb})^2 + (p_{za} - p_{zb})^2$$

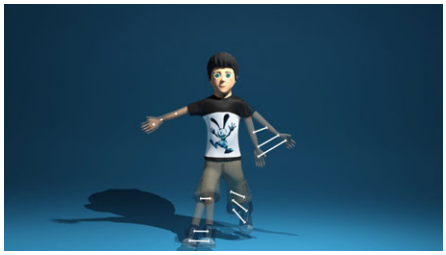
Existen diferente tipos de distancia entre esqueletos dependiendo de cómo se ponderen los pesos c_i: **distancia completa**, **distancia selectiva**, **distancia con umbral** y **distancia selectiva con umbral**. La distancia completa considera que todas las componentes del esqueleto contribuyen de la misma manera a la distancia. La distancia selectiva considera que sólo algunas componentes del esqueleto contribuyen a la distancia. La distancia con umbral establece un umbral mínimo de contribución en la suma total de distancia. Y la distancia selectiva con umbral une las características de la distancia selectiva y la distacia umbral. Más adelante se definirán cada una de las distancias existentes.

A continuación se muestran dos esqueletos correspondientes al movimiento de un gesto que utiliza las piernas y el brazo izquierdo. La distancia total es la suma de las rayas blancas ponderadas por su correspondiente peso.

(a) Esqueleto a



(b) Esqueleto b



(c) Distancia entre esqueletos

Figura 6.1: Distancia entre Esqueletos a y b

Fuente del Modelo 3D: TurboSquid. Modelo nº 708030 (Ver Notas en página III)

6.1.1. Distancia Completa

La distancia completa entre dos esqueletos E es la aplicación $d : E \times E \Rightarrow \mathbb{R}$ que suma equitativamente la distancia entre los puntos correspondientes de cada esqueleto.

$$d_n(a,b) = \sum_{i=1}^{20} \cdot d(p_{ia}, p_{ib}) \quad \forall\ a,b \in E$$

Aplicando la definición general de distancia:

$$d_n(a,b) = d(a,b)\Big|_{c_i=1}$$

En el ejemplo, la distancia completa es la suma de todas las rayas blancas.

Figura 6.2: Distancia Completa

6.1.2. Distancia Selectiva

La distancia selectiva entre dos esqueletos E es la aplicación $d : E \times E \Rightarrow \mathbb{R}$ que suma de manera equitativa la distancia entre las componentes del esqueleto de un rango s.

$$d_s(a,b) = \sum_i d(p_{ia}, p_{ib}) \quad \forall\ i \in s$$

Se define el rango de un esqueleto $a \in E$ como el conjunto de elementos de dicho esqueleto involucrados en el movimiento de un gesto.

$$s_a = (p_{a_1}, p_{a_2}, ..., p_{a_n}) \quad \forall\ p \in a$$

Aplicando la definición general de distancia:

$$d_s(a,b) = d(a,b)\Big|_{c_i=1,\ c_j=0} \quad \forall\ i\ /\ p_i \in s,\ j\ /\ p_j \notin s$$

Para la realización de gestos con las manos, se define un rango de selección s formado por los puntos de los codos, muñecas y manos.

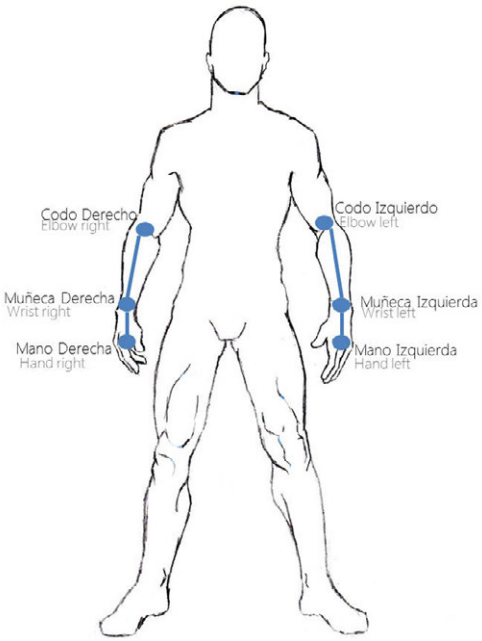


Figura 6.3: Rango de Selección

En el ejemplo, la distancia selectiva del rango s (manos, muñecas y codos) es sólo la suma de las líneas blancas correspondientes a esas zonas.

Figura 6.4: Distancia Selectiva

6.1.3. Distancia con Umbral

La distancia con umbral entre dos esqueletos E es la aplicación $d : E \times E \Rightarrow \mathbb{R}$ que suma de manera equitativa las distancias entre las componentes del esqueleto que superen un umbral u_d.

$$d_u(a,b) = \sum_i d(p_{ia}, p_{ib}) \quad \forall\ i \in [1,20]\ /\ d(p_{ia}, p_{ib}) > u_d$$

Aplicando la definición general de distancia:

$$d_u(a,b) = d(a,b)\Big|_{c_i=1,\ c_j=0} \quad \forall\ i \in [1,20]\ /\ d(p_{ia}, p_{ib}) > u_d,\ j \in [1,20] - i$$

La distancia con umbral es útil para eliminar el ruido provocado por las partes del cuerpo cuyo movimiento no forma parte del gesto. Aunque el ruido sea pequeño comparado con la contribución del movimiento, si en el movimiento sólo intervienen 2 partes del cuerpo, significa que otras 18 partes están introduciendo ruido. Y un ruido multiplicado por 18 ya no es despreciable.

En el ejemplo, la distancia con umbral es la suma de todas las rayas blancas excepto las más pequeñas que no superaran el umbral mínimo.

Figura 6.5: Distancia con Umbral

6.1.4. Distancia Selectiva con Umbral

La distancia selectiva con umbral entre dos esqueletos E es la aplicación $d : E \times E \Rightarrow \mathbb{R}$ que suma de manera equitativa la distancia entre las componentes del esqueleto de un rango s que superen un umbral u_d.

$$d_u(a,b) = \sum_i d(p_{ia}, p_{ib}) \quad \forall\ i \in s\ /\ d(p_{ia}, p_{ib}) > u_d$$

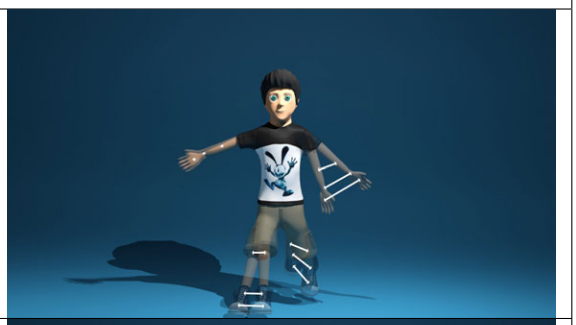
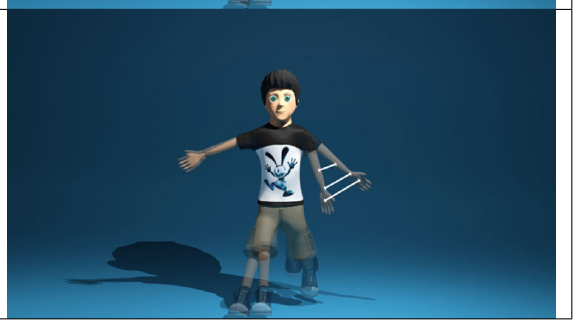
Aplicando la definición general de distancia:

$$d_u(a,b) = d(a,b)\Big|_{c_i=1,\ c_j=0} \qquad \forall\ i \in s\ /\ d(p_{ia}, p_{ib}) > u_d,\ \ j \in [1,20] - i$$

En el ejemplo, la distancia selectiva con umbral de codos, muñecas y codos es la suma de las líneas blancas correspondientes a esas zonas y que superen el umbral mínimo.

Figura 6.6: Distancia Selectiva con Umbral

6.1.5. Tabla Comparativa de Distancias

TIPO	DESCRIPCIóN	IMAGEN
DISTANCIA COMPLETA	Todas las componentes del esqueleto tienen el mismo peso	
DISTANCIA SELECTIVA	Sólo se tienen en cuenta ciertas componentes del esqueleto	
DISTANCIA CON UMBRAL	Sólo se tienen en cuenta las componentes cuyo movimiento supera un umbral	
DISTANCIA SELECTIVA CON UMBRAL	Sólo se tienen en cuenta ciertas componentes del esqueleto cuyo movimiento supere un umbral	

Cuadro 6.1: Tabla Comparativa de Distancias

De la definición de distancia entre esqueletos, se deduce que la distancia entre dos esqueletos $d(a, b)$ será pequeña (bajo coste) si el esqueleto a y b están en posiciones parecidas. Si están en posiciones diferentes, la distancia será grande (alto coste).

En el siguiente ejemplo se muestra el mismo instante de tiempo de dos gestos iguales: levantar la mano izquierda. El movimiento de la derecha se realiza un poco más rápido, por lo que la mano está más arriba que en el movimiento de la izquierda. Al ser el mismo gesto, la distancia entre un esqueleto y otro es muy pequeña.

(a) Esqueleto de un gesto (b) Esqueleto del mismo gesto

Figura 6.7: Esqueleto del mismo gesto

Figura 6.8: Distancia etre esqueletos de un mismo gesto

Por definición de distancia completa entre esqueletos, la distancia entre los dos esqueletos será la suma de las distancias entre cada componente de los esqueletos, es decir, la suma de los módulos de las flechas blancas del dibujo.

En este otro ejemplo se muestra el mismo instante de tiempo de dos gestos distintos: en el de la izquierda se levanta la mano izquierda, y en el de la derecha se levanta la mano derecha. Al ser dos gestos distintos, la distancia entre un esqueleto y otro es bastante más grande que en el ejemplo anterior.

(a) Esqueleto de un gesto (b) Esqueleto de otro gesto distinto

Figura 6.9: Esqueletos de gestos distintos

Figura 6.10: Distancia entre esqueletos de gestos distintos

Comparando las 2 distancias en un gráfico, se observa que el coste de alinear dos gestos distintos (segundo ejemplo) es mucho mayor que el coste de alinear dos gestos iguales (primer ejemplo).

Calculando la distancia entre cada par de elementos de las secuencias de esqueletos X e Y , se obtiene la matriz de costes $C \in \mathbb{R}^{N \times M}$ definida como $C(n, m) := d(x_n, y_m)$. El objetivo del algoritmo DTW es encontrar el alineamiento entre las secuencias de esqueletos X e Y con el mínimo coste.

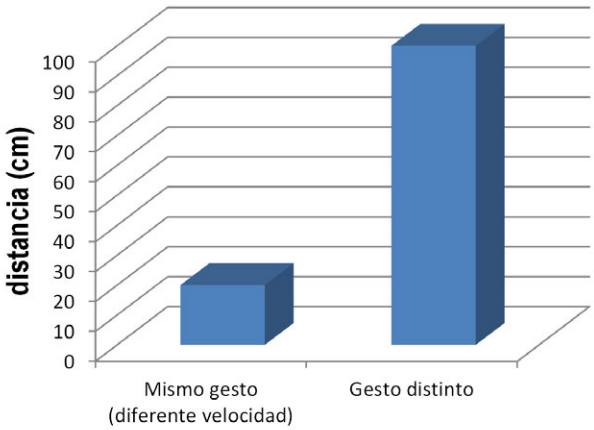


Figura 6.11: Grafica comparativa de distancias: mismo gesto y gestos distintos

6.2. Camino de Alineamiento

Definición Un (N,M)-camino de alineamiento[7] es una secuencia $p = (p_1, p_2 \ldots, p_n)$ / $p_i = (n_i, m_i) \in [1, N] \times [1, M]$ $\forall\ i \in [1, L]$ que cumple las siguientes condiciones:

1. Condición de frontera: $p_1 = (1, 1)\ \wedge\ p_L = (N, M)$
2. Condición de monotonicidad: $n_1 \leq n_2 \leq \ldots \leq n_L\ \wedge\ m_1 \leq m_2 \leq \ldots \leq m_L$
3. Condición de desplazamiento: $p_{l+1} - p_l \in \{(1, 0), (0, 1), (1, 1)\}\ \ \forall\, l \in [1, L-1]$

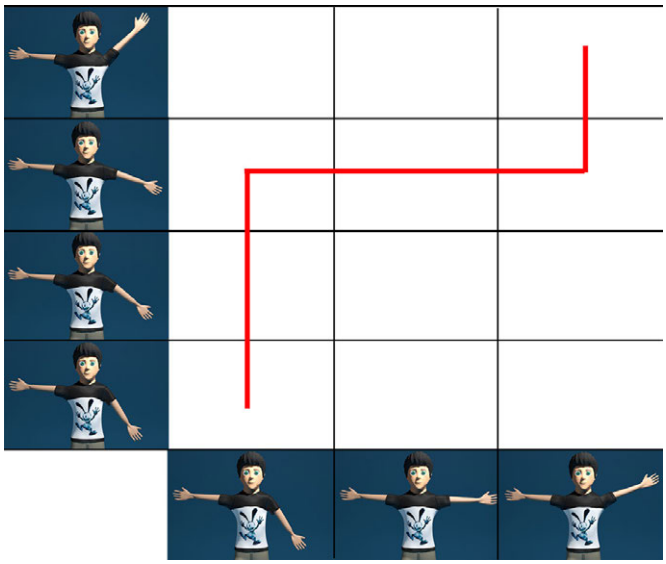


Figura 6.12: Camino de alineamiento de 2 gestos

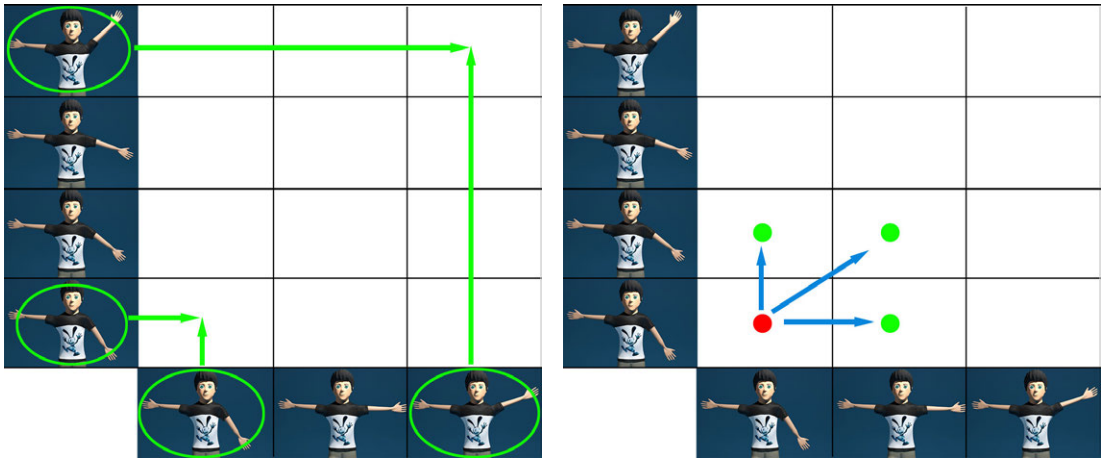
La condición de frontera impone que el primer punto del camino sea el punto formado por el primer elemento de las 2 secuencias a comparar, y el último punto del camino sea el punto formado por el último elemento de las 2 secuencias a comparar.

En términos de alineamiento esto se traduce a que los primeros elementos deben ser alineados como el primer punto del camino, y los últimos elementos de las 2 secuencias deben ser alineados como el último punto del camino.

La condición de monotonicidad realmente es una consecuencia de la condición de salto, ya que al avanzar de n_1 a n_2 o de m_1 a m_2 sólo puede variar 0 o 1. En efecto, sea $p_1 = (n_1, m_1)$, $p_2 = (n_2, m_2)$, por la condición de salto:

$$p_2 = (n_2, m_2) = \begin{cases} (n_1 + 1, m_1) \\ (n_1, m_1 + 1) \\ (n_1 + 1, m_1 + 1) \end{cases} \quad \Rightarrow \quad \begin{cases} n_1 \leq n_2 \leq \ldots \leq n_L \\ m_1 \leq m_2 \leq \ldots \leq m_L \end{cases}$$

La condición de desplazamiento establece que ningún elemento de las 2 secuencias puede ser ni omitido ni repetido.



(a) Camino de alineamiento: 1ª Condición (b) Camino de alineamiento: 3ª Condición

Figura 6.13: Condiciones Camino de alineamiento

La distancia recorrida por un camino de alineamiento entre las secuencias de esqueletos X e Y, definida la distancia entre dos esqueleto como d(x,y), es: $c_p(X, Y) := \sum_{l=1}^{L} d(x_{n_l}, y_{m_l})$.

Definición Se define el **camino de alineamiento óptimo** entre dos secuencias de esqueletos X e Y como el camino de alineamiento $p*$ que recorre la mínima distancia de entre todos los caminos de alineamiento posibles.

Definición Se define la *distancia DTW* entre 2 secuencias de esqueletos DTW(X,Y) como la distancia recorrida por el camino de alineamiento óptimo $p*$:

$$DTW(X,Y) := d_{p*}(X,Y) = min\,\{d_p(X,Y) \;\; / \;\; \text{p es un (N,M)camino de alineamiento}\}$$

Aunque se denomine *distancia DTW* , no es una distancia, ya que no cumple la condición de desigualdad triangular que debe cumplir una distancia.

Calcular el camino óptimo p* entre dos secuencias se reduce a hallar el camino con la distancia más corta. Una solución a este problema sería probar todos los caminos de alineamiento entre X e Y, y elegir el camino con la distancia más corta. Sin embargo, existen soluciones más óptimas mediante programación dinámica.

Sean las secuencias prefijo $X(1:n) := (x_1,\ldots,x_n) \;\; \forall \;\; n \in [1:N]$ e $Y(1:m) := (y_1,\ldots,y_m) \;\; \forall m \in [1:M]$, se define la matriz de distancia acumulada como $D(n,m) := DTW(X(1:n), Y(1:m))$

Teorema La matriz de distancia acumulada D cumple las siguientes igualdades:

$$\begin{cases} D(n,1) = \sum_{k=1}^{n} c(x_k, y_1) \;\; \forall \;\; n \in [1:N] \\ D(1,m) = \sum_{k=1}^{m} c(x_1, y_k) \;\; \forall \;\; n \in [1:M] \\ \boxed{D(n,m) = min\,\{D(n-1,m-1), D(n-1,m), D(n,m-1)) + c(x_n,y_m)\}} \\ \qquad \forall \;\; 1 < n \leq N \;\; \wedge \;\; \forall \;\; 1 < m \leq M \end{cases}$$

Algoritmo El algoritmo para calcular el camino de alineamiento óptimo sigue el siguiente procedimiento:

Entrada: Matriz de distancias acumuladas

Salida: Camino de alineamiento óptimo p*

Procedimiento: El camino óptimo $p* = (p_1,\ldots,p_L)$ es calculado en orden inverso empezando por $p_L = (N,M)$.

$$p_{l-1} := \begin{cases} (1, m-1), & \text{si n} = 1 \\ (n-1, 1), & \text{si m} = 1 \\ argmin\,\{D(n-1, m-1), D(n-1, m), D(n, m-1)\} & \text{en el resto} \end{cases}$$

Este algoritmo es el algoritmo básico para el cálculo de la distancia óptima, pero hay otras variantes que optimizan más aún el cálculo de la distancia DTW y que controlan el camino a seguir. Dichas variantes han sido implementadas en el sistema de reconocimiento de gestos, por lo que se explicarán en el siguiente apartado.

En el siguiente ejemplo se muestra el camino óptimo entre 2 secuencias de gestos.

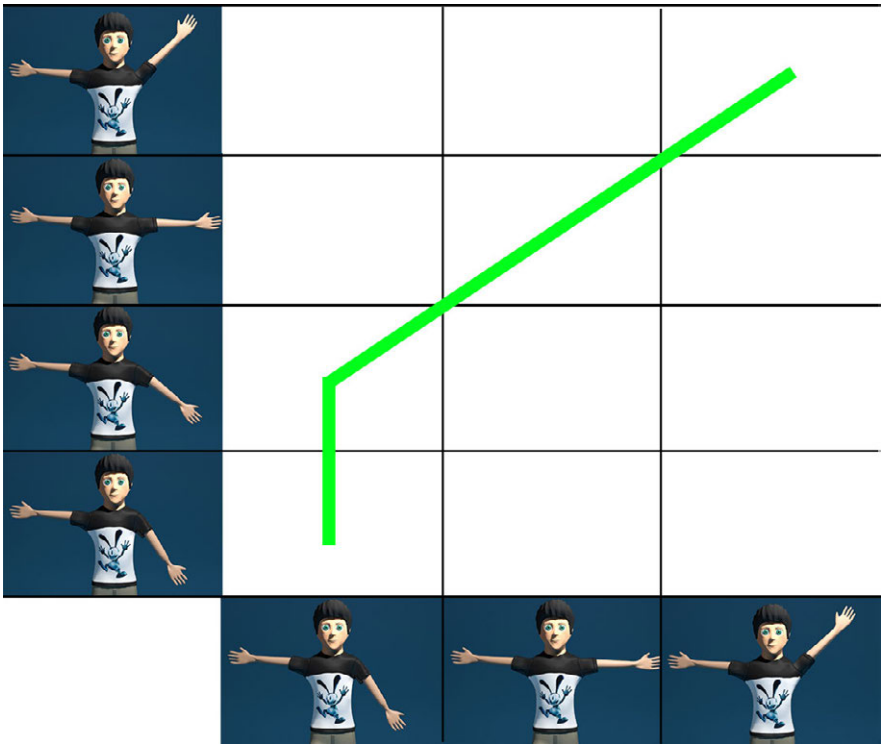


Figura 6.14: Camino de alineamiento óptimo de 2 gestos

6.3. Variaciones de DTW

En el algoritmo DTW se pueden introducir variaciones con el objetivo de optimizar y contolar la ruta del camino de alineamiento.

6.3.1. Condición Frontera en Inicio y Fin

En el algoritmo DTW establecimos como primera condición la condición de frontera: $p_1 = (1,1) \ \wedge \ p_L = (N,M)$. Esto significa que las dos secuencias se van a alinear completas, lo que puede es imprescindible al comparar dos gestos. Si por ejemplo tenemos un gesto que sube y baja el brazo, y otro que sólo baja el brazo, si no imponemos la condición frontera en (1,1) el gesto sería el mismo desde el punto de vista del algoritmo DTW, ya que alinearía la parte del bajar el brazo del primer gesto con todo el segundo gesto.

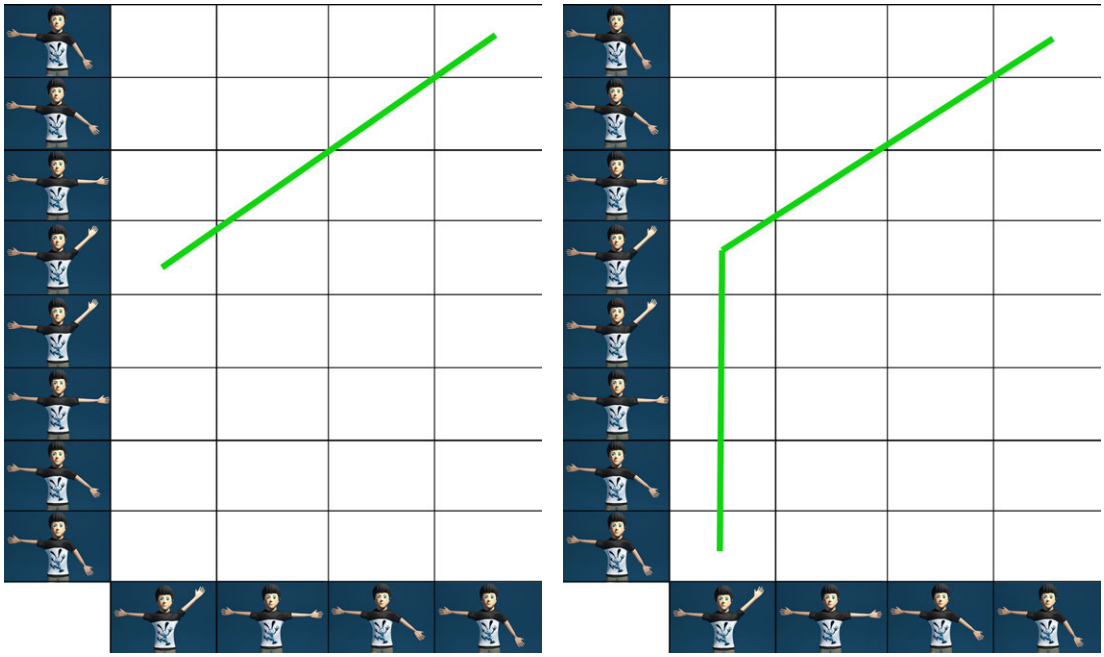
La condición frontera se impondrá en los casos en que sea necesario alinear las secuencias completas y no se considere lo mismo una misma subsecuencia que una secuencia. Sin embargo, hay casos en lo que no es necesario alinear las secuencias completas, y una subsecuencia se considera lo mismo que una secuencia completa. Por ejemplo, supongamos que tenemos una aplicación que nos identifica que parte del cuerpo hemos movido. Podemos comparar gestos de subir y bajar mano, piernas y brazos y comparar con el gesto realizado. Si el gesto es subir la mano derecha, nos interesa que se considere igual que el gesto de subir y bajar la mano derecha: cuanto menos distancia halla entre ellos, más fácil será de considerarlo como gesto detectado cuando se compare con los otros gestos entrenados.

Como vemos en la figura, la distancia entre el gesto de levantar y bajar la mano, y el gesto de bajar la mano, es mínima si no imponemos la condición de frontera. En cambio, si consideramos la condición de frontera, la distancia entre los gestos aumenta bastante.

En el evaluador de gestos se ha implementado la opción de habilitar o deshabilitar la condición frontera en el Inicio y en el Final. Por defecto está habilitado, y se recomienda mantenerlo así excepto que se quiera considerar que un gesto y una parte de ese gesto sean considerados iguales.

6.3.2. Condición de Desplazamiento

La tercera condición, la condición de desplazamiento, impone que cada esqueleto de la secuencia X tiene que estar alineado con un esqueleto de la secuencia



(a) Sin Condición de Frontera (b) Con Condición de Frontera

Figura 6.15: Camino de alineamiento con y sin Condiciones de Frontera

Y. Sin embargo, hay situaciones en la que un mismo esqueleto se alinea con una serie de esqueletos de la otra secuencia, dando lugar a largos caminos horizontales o verticales. Esto puede provocar que el camino de alineamiento se quede bloqueado en un cierto esqueleto de una secuencia dando lugar a un caminos degenerados.

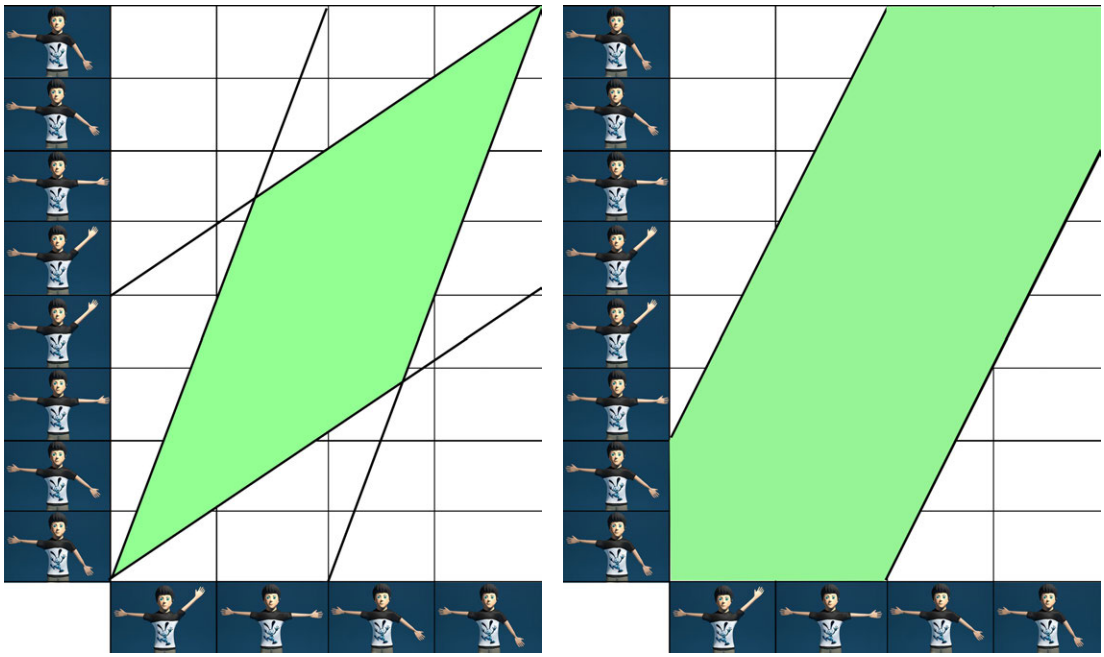
Paralelogramo de Itakura

El paralelogramo de Itakura define una región (paralelogramo) por donde debe ir el camino de alineamiento, restringiendo la pendiente del camino. Esta región está definida por un desplazamiento horizontal máximo y un desplazamiento diagonal.

Banda de Sakoe-Chiba

La Banda de Sakoe-Chiba define una región por donde debe ir el camino de alineamiento, restringiendo el desplazamiento horizontal y vertical. Esta región está definida por una pendiente.

En el evaluador de gestos se han implementado tanto el Paralelogramo de Itakura como la Banda de Sakoe-Chiba. En el apartado de configuración del algoritmo DTW se pueden establecer los valores para los desplazamientos que definen el *Paralelogramo de Itakura*, y la pendiente de la *Banda de Sakoe-Chiba*.



(a) Paralelogramo de Itakura (b) Banda de Sakoe-Chiba

Figura 6.16: Condiciones de Desplazamiento: Paralelogramo de Itakura y Banda de Sakoe-Chiba

Capítulo 7

Software: Evaluador de Gestos

El evaluador de gestos es una aplicación gráfica en Windows que permite detectar gestos mediante un sensor Kinect. La aplicación se comunica con una base de datos en MySQL, donde se guardan todos los datos de la aplicación: configuración, gestos grabados, evaluaciones realizadas... Toda la lógica de la detección de gestos se ha implementado en una librería llamada *Algorithms*, con el fin de que pueda ser utilizada por otras aplicaciones, como el *Teleoperador de Robot Mindstorm*, que veremos en la Segunda parte de la memoria.

Figura 7.1: Esquema de la Aplicación

Tecnología La solución está programada en el lenguaje C#, usando Windows Presentation Foundation (WPF), con .Net Framework 4.5. La base de datos está implementada mediante Entity Framework siguiendo el enfoque Code First: se crea un modelo de clases POCO (Plain Old CLR Object) a partir del cual se puede generar una base de datos y/o mapear esas clases a una base de datos existente.

Librerías Externas La solución hace uso de las siguientes librerías externas:

- **EntityFramework**: Framework que permite trabajar con elementos de la base de datos (tablas) usando objetos. Elimina la necesidad de la mayor parte del código de acceso a datos.
- **Helix 3D Toolkit**: Colección de controles y clases auxiliares para realizar manejar objetos en 3D en WPF.
- **Conector MySQL**: Driver para C# que permite conectar una aplicación con una base de datos MySQL.
- **TCD.Kinect**: Librería que permite mostrar un esqueleto en un espacio de 3 dimensiones
- **WPF Animated GIF**: Librería que permite reproducir una imagen GIF
- **MoreLINQ**: Añade métodos a LINQ to Objects.

Estructura La solución del evaluador de gestos está dividido en 4 proyectos:

- *Algorithms* - Librería interna que sirve de soporte al Evaluador de Gestos.
- *GestureEvaluator* - Evaluador de Gestos. Hace uso de la librería interna (Algorithms) y de la base de datos (Models).
- *Models* - Base de datos.

7.1. Librería - (*Algorithms*)

En términos generales, podemos decir que el **Evaluador de Gestos** contiene toda la parte gráfica, el **Modelo** contiene la estructura de la base de datos, y la **Librería** contiene toda la lógica del Evaluador de Gestos relativa a los gestos:

- Algoritmo DTW
- Distancia entre esqueletos
- Gestos
- Grabador y reproductor de gestos
- Preprocesador de esqueletos
- Reconocedor de Gestos

La estructura de la solución no sigue exactamente un patrón MVC (Model-View-Controller) como puede parecer a priori. En el patrón MVC, el flujo de la aplicación es manejado por el controlador que interactúa con la base de datos a través del Modelo y muestra las páginas usando plantillas (Vista). Sin embargo, en la solución empleada, todo el flujo del Evaluador de Gestos es manejada por el propio Evaluador de Gestos, que llama al Modelo cuando necesita interactuar con la base de datos o llama a la Librería cuando necesita realizar alguna lógica relativa a los gestos.

Figura 7.2: Estructura del Evaluador de Gestos

Como veremos en el propio *Evaluador de Gestos*, y en el *Controlador de Robot Mindstorm* en la segunda parte de la memoria, la librería puede ser usada por cualquier aplicación. Para ello es necesario importar la .dll generada por el proyecto Algorithms, *Algorithms.dll*. Una vez importada la librería, se puede hacer uso de las clases y métodos que veremos a continuación.

7.1.1. Algoritmo DTW - *Dtw*

La clase DTW implementa el algoritmo DTW para calcular la distancia de alineamiento entre 2 gestos. Esta clase es una modificación de la clase DTW de la aplicación NDTW: Dynamic Time Warping (DTW) algorithm implementation for .NET C#. Esta aplicación ha sido desarrollada por Darjan Oblak, publicada bajo licencia MIT, y disponible en https://github.com/doblak/ndtw.

El uso de esta clase DTW nos ha permitido incorporar un algoritmo DTW implementado de una manera eficiente para el lenguaje C# y donde existe una comunidad que ha comprobado que dicha implementación no contiene errores. Una de las premisas de este proyecto ha sido utilizar las ventajas de librerías o aplicaciones con licencia que permitan su uso en aplicaciones académicas.

La clase DTW de la aplicación NDTW compara 2 secuencias de datos unidimensionales, por lo que ha sido necesario modificar esta clase para comparar 2 gestos, es decir, 2 secuencias de esqueletos. Cada esqueleto está compuesto de 20 puntos del esqueleto tridimensionales:

NDTW	DTW
2 secuencias de puntos	2 secuencias de esqueletos
1 punto = 1 número	1 esqueletos = 20 puntos tridimensionales

La modificación de la clase DTW no ha sido muy compleja, ya que simplemente se ha modificado el cálculo de las distancias: donde antes se calculaba la distancia entre 2 puntos de la secuencia, ahora se calcula la distancia entre 2 esqueletos.

Este algoritmo DTW permite utilizar:

- Condiciones de Frontera en (1, 1) y (m,n)
- Condiciones de desplazamiento:
  - Máxima pendiente (Sakoe-Chiba)
  - Máximo desplazamiento horizontal y vertical (Ikatura)

La clase *Dtw* contiene las siguientes propiedades:

Gesture x	Primer Gesto
Gesture y	Segundo Gesto para comparar con el primero.
isXLongerOrEqualThanY	Si el gesto X tiene más esqueletos que Y
signalsLengthDifference	Número de esqueletos de diferencia entre los 2 gestos.
distance	Tipo de distancia (Normal = 1, Selective = 2, Threshold = 3, SelectiveWithThreshold = 4
boundaryConstraintStart	Condición de frontera en (1, 1)
boundaryConstraintEnd	Condición de frontera en (m, n)
sakoeChibaConstraint	Indica si se aplica la banda Sakoe-Chiba
sakoeChibaMaxShift	Máximo desplazamiento en la banda Sakoe-Chiba
calculated	Indica si la distancia entre los 2 gestos ha sido calculada.
distances	Matriz de distancias, D. Dado dos esqueletos, $A \in X$, $B \in Y$, D[A,B] es la distancia entre el esqueleto A y el esqueleto B.
pathCost	Matriz de costes, C. Dado dos instantes, $a, b, C[a, b]$ es el coste de alinear los gestos X (hasta el instante m - a) e Y (hasta el instante n - b). Así el valor de C[0,0] es el coste de alinear los 2 gestos X e Y.
useSlopeConstraint	Indica si se limita la pendiente del desplazamiento
slopeMatrixLookbehind	Indica cuanto se incrementa la matriz dtw. En caso de no utilizar condiciones de saltos, esta variable vale 1. En caso contrario, vale 1 + salto lateral + salto diagonal.
slopeStepSizeDiagonal	Máximo desplazamiento en diagonal (forma el paralelogramo de Itakura)
slopeStepSizeAside	Máximo desplazamiento lateral (forma el paralelogramo de Itakura)
_predecessorStepX	Variable interna que indica el desplazamiento en X para una X e Y dadas
_predecessorStepY	Variable interna que indica el desplazamiento en Y para una X e Y dadas

La clase *Dtw* contiene los siguientes métodos:

GetCostMatrix	Calcula la matriz de costes.
GetDistanceMatrix	Calcula la matriz de distancias.
GetPath	Calcula el camino de distancias.
GetCost	Calcula el camino de costes.
Calculate	Calcula las matrices de distancia y coste.
CalculateWithSlopeLimit	Calcula la matriz de coste con condiciones de desplazamiento.
CalculateWithoutSlopeConstraint	Calcula la matrix de coste sin condiciones de desplazamiento.
CalculateDistances	Calcula la matriz de distancias.
InitializeArrays	Inicializa las matrices.

7.1.2. Distancia entre esqueletos - *Distances*

La clase *Distances* implementa el cálculo de la distancia entre 2 esqueletos. Existen 4 tipos de distancia:

1. **Completa** - Todas las componentes del cuerpo aportan de la misma manera a la distancia.

2. **Selective** - Sólo ciertas componentes del cuerpo aportan a la distancia.

3. **Threshold** - Sólo las componentes del cuerpo cuya distancia supere el umbral contribuyen a la distancia.

4. **SelectiveWithThreshold** - Sólo ciertas componentes del cuerpo cuya distancia supere el umbral contribuyen a la distancia.

   Además, se han definido 4 distancias del tipo *Selective*:

5. ***SelectiveUpperBody*** - Sólo los hombros, codos, muñecas y manos contribuyen a la distancia.

6. ***SelectiveArms*** - Sólo los codos, muñecas y manos contribuyen a la distancia.

7. ***SelectiveUpperBodySquared*** - Sólo los hombros, codos, muñecas y manos contribuyen a la distancia, siendo esta la suma de las distancias al cuadrado de cada componente.

8. ***SelectiveUpperBodyCube*** - Sólo los hombros, codos, muñecas y manos contribuyen a la distancia, siendo esta la suma de las distancias al cubo de cada componente.

La explicación de cada tipo de distancia está detallada la sección Distancia entre esqueletos, en la página 56.

La clase *Distances* contiene las siguientes propiedades:

EuclideanDistance	Calcula la distancia euclídea entre dos componentes del esqueleto. Cada componente del esqueleto es un punto en 3 dimensiones.
NormalDistance	Calcula la distancia completa entre dos esqueletos.
SelectiveDistance	Calcula la distancia selectiva entre dos esqueletos.
SelectiveDistancePow	Calcula la distancia selectiva entre dos esqueletos. La distancia es la suma de las distancias de cada componente elevada a una potencia.
SelectiveDistanceWithThreshold	Calcula la distancia selectiva con umbral entre dos esqueletos..
SelectiveDistanceWithThresholdAndPow	Calcula la distancia selectiva con umbral entre dos esqueletos. La distancia es la suma de las distancias de cada componente elevada a una potencia.
ThresholdDistance	Calcula la distancia umbral entre dos esqueletos.

La clase *Distances* contiene los siguientes métodos:

Joints	Componentes del esqueleto (sólo empleados en las distancias selectivas).
Type	Tipo de distancia.
Threshold	Valor del umbral (sólo empleado en las distancias con umbral).

7.1.3. Gesto - *Gesture*

La clase *Gesture* representa a un gesto, que por definición es una secuencia de esqueletos. Esta clase implementa los métodos de grabar y reproducir un gesto, para lo cual implementa la máquina de estados genérica descrita en la página 32.

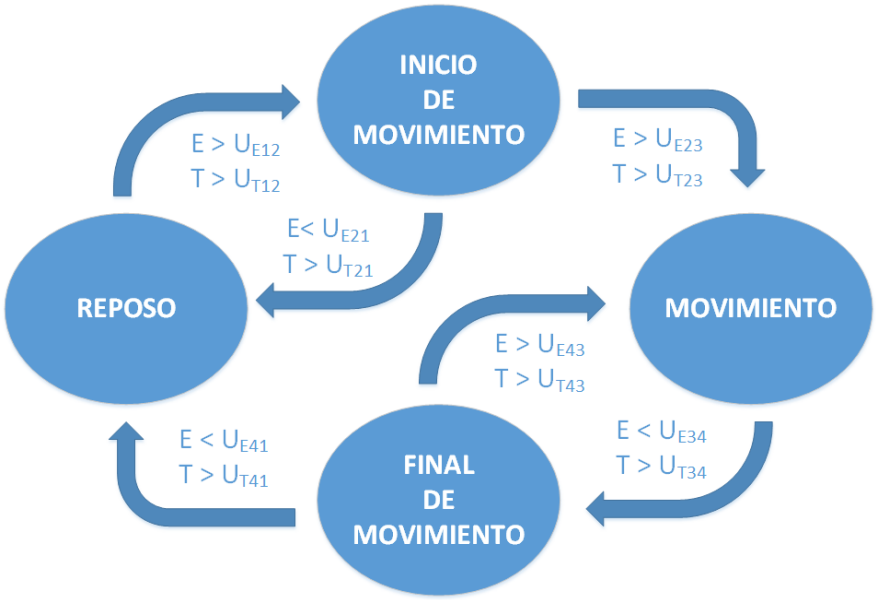


Figura 7.3: Máqina de Estados implementada

Cuando se llama al método de grabar un gesto, se crea el archivo donde se va a guardar el gesto y se añade el gesto en la base de datos. En la base de datos sólo se guarda la información relativa al usuario, tipo de gesto (entrenamiento o evaluación) y ruta del archivo donde está el gesto. En el archivo se guarda por cada frame grabado: el timestamp, el plano del suelo y el esqueleto serializado. En el siguiente apartado, Grabador y reproductor de gestos, se especifica con más detalles el proceso de grabación de gestos.

Cuando queremos grabar un gesto, realizamos el siguiente proceso:

1. Llamamos al método Record de la clase gesto para crear el archivo y guardar el gesto en base de datos.

2. Por cada frame que recibimos, llamamos al método Record(SkeletonFrame). Para ello, asociamos al evento del sensor de que hay un nuevo frame con el método Record(SkeletonFrame):

   sensor.SkeletonFrameReady += sensor_SkeletonFrameReadyRecord;

Internamente en la clase gesto, la grabación de frames no se realiza continuamente desde que se invoca al método de grabar, sino que sigue el siguiente esquema:

1. Al llamar al método Record, se crea el archivo y se guarda la información del gesto en la base de datos.

2. Al llamar al método Record(SkeletonFrame), cada vez que hay un nuevo frame pasa por la máquina de estados:

   - Si está en estado *Reposo*, no se graba el frame en el archivo.

   - Si está en estado *Inicio de Movimiento*, *Movimiento* o *Fin de Movimiento*, se graba el frame en el archivo

     recorder.Record(frame);

La clase *Gesture* tiene las siguientes propiedades:

ID	Identificador del gesto en la base de datos.
gesturename	Nombre del gesto. Si es un gesto de entrenamiento, correponde con la descripción del gesto realizado. Si es un gesto de evaluación, es el timestamp (fecha y hora) de cuando se realizó el gesto.
username	Usuario que realizó el gesto.
Length	Número de esqueletos que componen el gesto.
Skeletons	Esqueletos que componen el gesto.
recorder	Grabador de gestos. Se utiliza para grabar el gesto.
replay	Reproductor de gestos. Se utiliza para reproducir el gesto.

state	Estado actual de la máquina de estado.
energy	Energía actual del movimiento.
lastSkeleton	Último esqueleto. Utilizado para calcular la energía del movimiento como la distancia entre el esqueleto actual y el último esqueleto.
energies	Array de energías de movimiento.
ctx	Conector con la base de datos
ET_RestingToStartMoving	Umbral de energía para pasar del estado de Reposo a Inicio de Movimiento.
ET_StartMovingToResting	Umbral de energía para pasar del estado de Inicio de Movimiento a Reposo.
ET_StartMovingToMoving	Umbral de energía para pasar del estado de Inicio de Movimiento a Movimiento.
ET_MovingToStopMoving	Umbral de energía para pasar del estado de Movimiento a Fin del Movimiento.
ET_StopMovingToMoving	Umbral de energía para pasar del estado de Fin del Movimiento a Movimiento.
ET_StopMovingToResting	Umbral de energía para pasar del estado de Fin del Movimiento a Reposo.
TT_RestingToStartMoving	Umbral de tiempo para pasar del estado de Reposo a Inicio de Movimiento.
TT_StartMovingToResting	Umbral de tiempo para pasar del estado de Inicio de Movimiento a Reposo.
TT_StartMovingToMoving	Umbral de tiempo para pasar del estado de Inicio de Movimiento a Movimiento.
TT_MovingToStopMoving	Umbral de tiempo para pasar del estado de Movimiento a Fin del Movimiento.
TT_StopMovingToMoving	Umbral de tiempo para pasar del estado de Fin del Movimiento a Movimiento.
TT_StopMovingToResting	Umbral de tiempo para pasar del estado de Fin del Movimiento a Reposo.
TC_RestingToStartMoving	Contador de tiempo para pasar del estado de Reposo a Inicio de Movimiento.
TC_StartMovingToResting	Contador de tiempo para pasar del estado de Inicio de Movimiento a Reposo.

TC_StartMovingToMoving	Contador de tiempo para pasar del estado de Inicio de Movimiento a Movimiento.
TC_MovingToStopMoving	Contador de tiempo para pasar del estado de Movimiento a Fin del Movimiento.
TC_StopMovingToMoving	Contador de tiempo para pasar del estado de Fin del Movimiento a Movimiento.
TC_StopMovingToResting	Contador de tiempo para pasar del estado de Fin del Movimiento a Reposo.
movementDetectionDistance	Calcula las distancias entre dos esqueletos.
movementDetectionPreprocesor	Preprocesador de gestos.

La clase *Gesture* tiene los siguientes métodos:

Delete	Borra el gesto de la base de datos y el archivo correspondiente.
DeleteSkeletonFile	Borra el archivo del gesto.
EvaluationRecord	Graba un gesto del tipo evaluación.
InitializeConf	Inicializa la configuración: preprocesador de esqueleto y umbrales de tiempo y distancia para la máquina de estados.
Record(SkeletonFrame frame)	Graba un frame del gesto.
Record(GestureType gesturetype)	Crea el gesto en la base de datos y crea el archivo correspondiente.
Replay	Reproduce un gesto.
ResetCounters	Resetea los contadores asociados a las transiciones de estados en la máquina de estados: ▪ TC_RestingToStartMoving ▪ TC_StartMovingToResting ▪ TC_StartMovingToMoving ▪ TC_MovingToStopMoving ▪ TC_StopMovingToMoving ▪ TC_StopMovingToResting
TrainingRecord	Graba un gesto del tipo entrenamiento.

7.1.4. Grabador y reproductor de gestos

Grabador de gestos

El grabador de gestos se encarga de grabar en un fichero los frames correspondientes a un gesto. Está compuesto por 2 clases:

- *KinectRecorder*
- *SkeletonRecorder*

Los frames se mandan al grabador con el método *Record(frame)*. La clase *SkeletonFrame* contiene los siguiente parámetros:

DepthFilter	Reservada para usos futuros.
FloorClipPlane	Plano del suelo en la forma de vector: (x, y, z, w).
FrameNumber	Número de frame.
SkeletonArrayLength	Número de esqueletos en el frame.
Timestamp	Fecha y hora del frame.
TrackingMode	Propiedad no recomendada usar. Su valor variará en el futuro.

Guardar toda la clase *SkeletonFrame* serializada podría ser una posibilidad, sin embargo, observamos que hay varios campos que no son necesarios: DepthFilter y TrackingMode. Además, el array de esqueletos no es una propiedad, sino que se obtiene a través del método *frame.CopySkeletonDataTo*. En lugar de guardar el objeto *SkeletonFrame*, guardamos la siguiente información:

- Número de milisegundos transcurridos desde el frame anterior al frame actual. Esto nos permitirá reproducir el gesto.
- Plano del suelo.
- Número de Frame (FrameNumber)
- Array de esqueletos. El array de esqueletos lo guardamos como un objeto serializado.

La clase *KinectRecorder* contiene las siguientes propiedades:

file	Archivo donde se guarda el gesto.
recordStream	Flujo de datos en memoria.
previousFlushDate	Fecha y hora de la última vez que se hizo un flush.
skeletonRecorder	Valor del umbral (sólo empleado en las distancias con umbral).
writer	Escribe tipos primitivos como valores binarios con su codificación específica.

La clase *KinectRecorder* contiene los siguientes métodos:

Flush	Método para asegurarnos que todos los datos en el buffer han sido enviados al archivo. Esta acción se realiza cada 60 segundos mientras se está grabando.
KinectRecorder	Constructor. Se le pasa como parámetro el archivo donde se va a guardar el gesto.
Record	Método para grabar un frame.
Restart	Reinicia la grabación. Este método es empleado cuando el gesto está grabando en el estado *Inicio de Movimiento*, pero vuelvo al estado *Reposo*.
Stop	Termina la grabación del gesto.

La clase *SkeletonRecorder* contiene las siguientes propiedades:

referenceTime	Fecha y hora de referencia. Para hallar los milisegundos que han pasado desde el último frame hasta el frame actual, se resta la fecha y hora actual al referenceTime.
writer	Escribe tipos primitivos como valores binarios con su codificación específica.

La clase *SkeletonRecorder* contiene los siguientes métodos:

Record	Método para grabar un frame.
Restart	Reinicia la grabación.

Reproductor de gestos - *KinectReplay*

El reproductor de gestos se encarga de reproducir un gesto a partir de los datos guardados en un fichero con los frames correspondientes al gesto. El reproductor está compuesto por 5 clases:

- *KinectReplay*
- *ReplayFrame*
- *ReplaySkeletonFrame*
- *ReplaySkeletonFrameReadyEventArgs*
- *ReplaySystem*

Para reproducir un gesto, se sigue el siguiente esquema:

1. Se crea la clase *KinectReplay*, y se invoca al método reproducir.
2. *KinectReplay* crea un objeto *ReplaySystem*, y le pasa todos los frames guardados en el archivo donde está guardado el gesto.
3. *ReplaySystem* lee un frame, espera el tiempo indicado en el campo TimeStamp (en milisegundos), y lanza el evento de que está listo un frame para ser reproducido.
4. Se continua realizando el paso 3 hasta que se termina el gesto.

Las clases *KinectReplay* y *ReplaySystem* modelan el comportamiento del reproductor a través de los métodos de reproducir (Start) y parar (Stop). A través de un objeto ReplaySystem, KinectReplay reproduce o para la reproducción de frames. KinectReplay es la clase que maneja el reproductor de frames, ReplaySystem.

La clase *ReplayFrame* es una clase abstracta que es heredada por *ReplaySkeletonFrame* y *ReplaySystem*. Estas clases que heredan las clase *ReplayFrame* incorporan las propiedades *FrameNumber* y *TimeStamp*.

La clase *ReplaySkeletonFrame* recupera, a partir del archivo del gesto, todos los campos guardados:

- Número de milisegundos transcurridos desde el frame anterior al frame actual.
- Plano del suelo.
- Número de Frame (FrameNumber)
- Array de esqueletos.

La clase *ReplaySkeletonFrameReadyEventArgs* se utiliza como la clase equivalente a *SkeletonFrameReadyEventArgs*. Son eventos lanzados cuando un frame está listo para ser reproducido.

La clase *KinectReplay* contiene las siguientes propiedades:

reader	Lee tipos de datos primitivos como valores binarios en una codificación específica.
skeletonReplay	Objeto ReplaySystem para reproducir frames.
stream	Flujo para conectar el BinaryReader con el archivo donde está guardado el gesto.
synchronizationContext	Proporciona la funcionalidad para lanzar los eventos cuando un frame está listo para ser reproducido.
Started	Variable para indicar si la reproducción ha comenzado.
IsFinished	Variable para indicar si la reproducción ha terminado.
SkeletonFrameReady	Evento que se lanza cuando un frame está listo para ser reproducido.

La clase *KinectReplay* contiene los siguientes métodos:

Dispose	Liberar todos los recursos de la propiedad de KinectReplay.
Start	Inicia la reproducción.
Stop	Para la reproducción.

La clase *ReplaySystem* contiene los siguientes métodos:

AddFrame	Añade un frame a la lista de frames para ser reproducidos.
Start	Inicia la reproducción.
Stop	Para la reproducción.

La clase *ReplaySystem* contiene las siguientes propiedades:

cancellationTokenSource	Permite cancelar la reproducción de un gesto.
frames	Lista de frames del gesto que tienen que ser reproducidos.
FrameReady	Evento que se lanza cuando un frame está listo para ser reproducido.

La clase *ReplayFrame* contiene las siguientes propiedades:

FrameNumber	Permite cancelar la reproducción de un gesto.
TimeStamp	Lista de frames del gesto que tienen que ser reproducidos.

La clase *ReplayFrame* contiene los siguientes métodos:

CreateFromReader	Crea un frame a partir de la información leída del archivo donde está guardado el gesto. Este método extrae las propiedades: TimeStamp, FloorClipPlane, FrameNumber y Skeletons.

La clase *ReplaySkeletonFrame* contiene las siguientes propiedades:

FloorClipPlane	Plano del suelo.
Skeletons	Array de esqueletos en el frame.

La clase *ReplaySkeletonFrame* contiene los siguientes métodos:

CreateFromReader	Crea un frame a partir de la información leída del archivo donde está guardado el gesto. Este método extrae las propiedades: TimeStamp, FloorClipPlane, FrameNumber y Skeletons.

La clase *ReplaySkeletonFrameReadyEventArgs* contiene las siguientes propiedades:

SkeletonFrame	Objeto ReplaySkeletonFrame.

7.1.5. Preprocesador de esqueletos

El preprocesador de esqueleto puede realizar realizar tres tipos de procesamiento:

- Centrar un esqueleto
- Escalar un esqueleto
- Rotar un esqueleto
  - Mediante Cambios de Base
  - Mediante Giro sobre el eje Y

Todas las clases están basadas en la interfaz *IPreprocessor*, que contiene el método Preprocess que lo implementan todas las clases de preprocesado. La clase *NonePreprocessor* implementa un preprocesador que no realiza ningún procesado.

Considerando el siguiendo esqueleto:

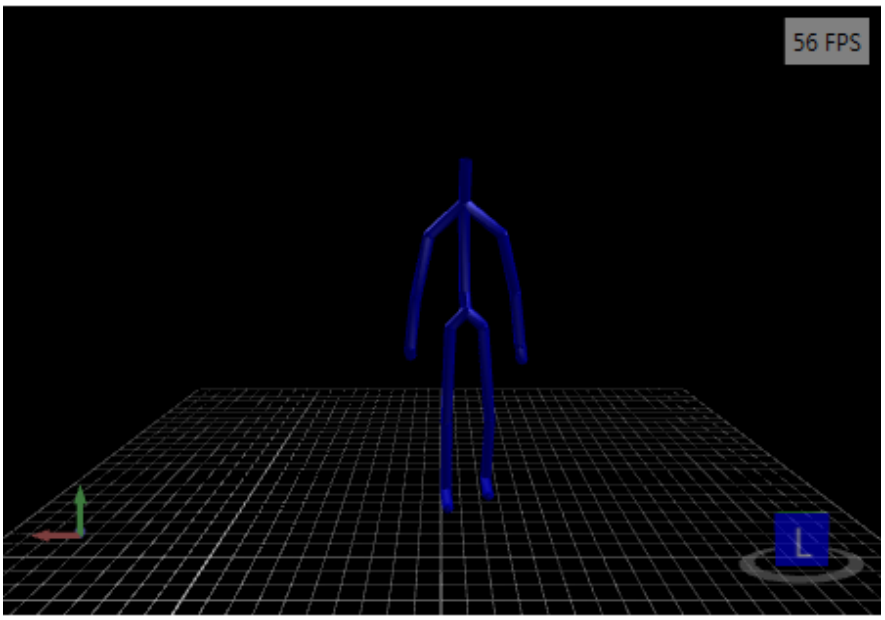


Figura 7.4: Esqueleto Original

A continuación se detallarán cada uno de los preprocesados implementados y la salida que produce cada uno de ellos sobre el esqueleto anterior.

Centrado - *CenterPreprocessor*

El centrado consiste en traslada el esqueleto situando el centro de la cadera en el punto (0,0,1):

1. **Se obtiene el vector de traslación como la resta del punto central de la cadera y el punto (0,0,1)**:

   $\vec{v_t} = (0 - c_x,\ 0 - c_y,\ 1 - c_z) = -(c_x,\ c_y,\ c_z - 1)$

2. **Se traslada cada punto sumado a cada punto el vector de traslación**:

   $p' = (p_x - c_x,\ p_y - c_y,\ p_z - c_z + 1)$

El punto (0,0,1) se ha escogido porque es el origen de coordenadas de la X e Y, y en el eje Z una distancia de 1 metro permite ver bien el esqueleto al ser reproducido. El plano Y = 0 no es el suelo, sino el plano paralelo al suelo a la altura del Kinect. En este punto estamos centrando el esqueleto en la misma coordenada X e Y donde está el Kinect, y a 1 metro de distancia (profundidad) del Kinect.

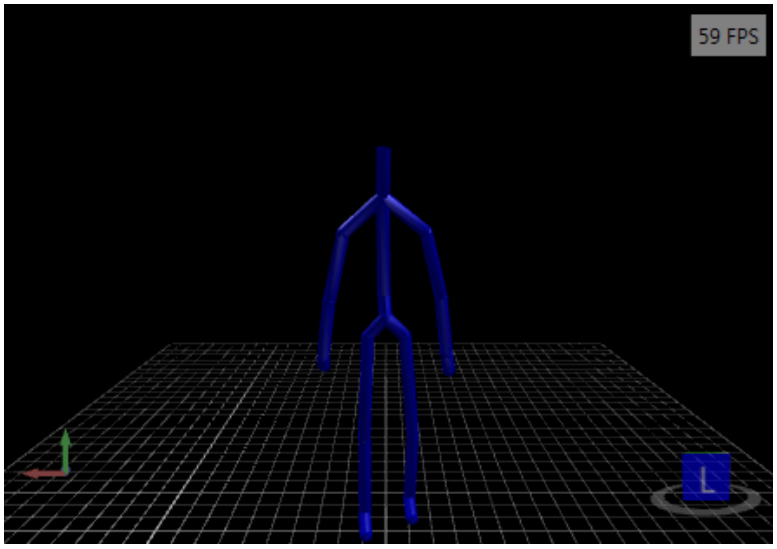


Figura 7.5: Esqueleto Centrado

La clase *CenterPreprocessor* tiene el siguiente método:

Preprocess	Centra el esqueleto.

Escalado - *ScalePreprocessor*

Este procesado escala el esqueleto para que siempre tenga el mismo tamaño. Como Kinect proporciona el esqueleto en coordenadas reales en 3 dimensiones, el escalado no tiene sentido ya que el esqueleto siempre tiene la misma proporción. Sin embargo, se ha añadido este preprocesado para comprobar los resultados del escalado en un esqueleto en coordenadas reales no tiene ninguna modificación.

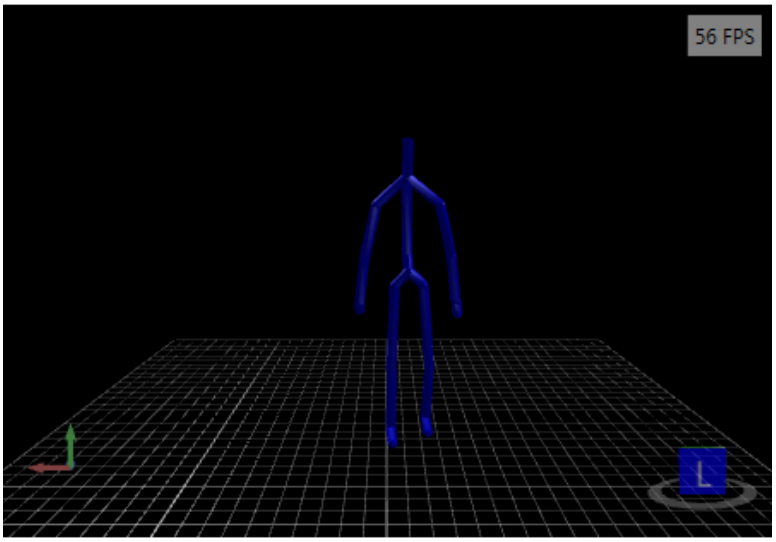


Figura 7.6: Esqueleto Escalado

La clase *ScalePreprocessor* tiene el siguiente método:

Preprocess	Escala el esqueleto.

Rotado

Para implementar la detección de un esqueleto girado, introducimos la rotación del esqueleto para que este esté siempre de frente. Para ello rotaremos todo el esqueleto para que la línea imaginaria que une los hombros quede paralela al eje X. La rotación se hará primero hallando el ángulo entre la línea imaginaria que une los hombros y el eje X, y posteriormente rotando todo el esqueleto este ángulo.

Existen 2 tipos de rotación:

- Rotación sobre el eje Y
- Cambio de base

RotatePreprocessor

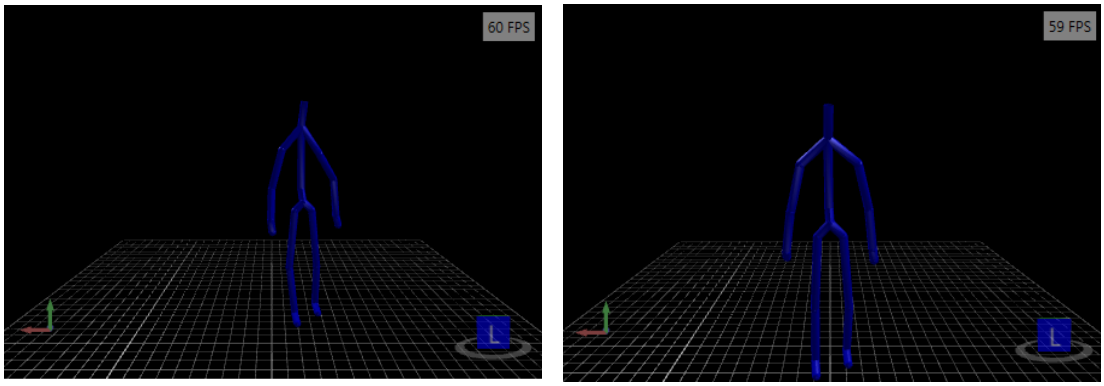
Se rota el cuerpo girando sobre el tronco, es decir, sobre el eje Y. El algoritmo de rotación implementado consiste en:

1. **Se obtiene el ángulo que forman la línea imaginaria que une los hombros y el eje X.**

   El ángulo se obtiene calculando el arcocoseno del producto escalar de la línea que une los hombros y el eje X.

2. **Se halla la matriz de rotación.**

3. **Se rota cada vector con la matriz de rotación.**



(a) Esqueleto Girado Original (b) Esqueleto Girado Rotado

Figura 7.7: Rotación mediante giro sobre el eje Y

ChangeBasePreprocessor

El cambio de base consiste en considerar siempre una misma base de referencia, que estará localizada en el propio esqueleto. De esta manera, la referencia a las componentes del esqueleto siempre es la misma.

Para calcular la matriz de cambio de base $M_{BB'}$:

1. Se obtienen los vectores de la base formada por el esqueleto:
2. Una vez obtenidos los vectores de la base, se forma la matriz de cambio de base B' a B, $M_{B'B}$, como la matriz cuyas columnas son los vectores de la base obtenidos.
3. Se calcula la matriz de cambio de base B a B', $M_{BB'}$, como la inversa de la matriz $M_{B'B}$

Base del Esqueleto Esta base está formada por:

$$\begin{cases} \text{Eje X – Recta que une los hombros.} \\ \text{Eje Y - Recta que une la cadera y el centro entre los hombros.} \\ \text{Eje Z – Perpendicular al eje X e Y.} \end{cases}$$

Sean $a = (a_x, a_y, a_z)$ el punto de la cadera, y sea $b = (b_x, b_y, b_z)$ el centro entre los hombros, $c = (c_x, c_y, c_z)$ el hombro derecho e $d = (d_x, d_y, d_z)$ el hombro izquierdo:

$$\begin{cases} \text{Eje X} = U = (c_x - d_x, c_y - d_y, c_z - d_z) \\ \text{Eje Y} = V = (a_x - b_x, a_y - b_y, a_z - b_z) \\ \text{Eje Z} = W = U \times V \end{cases}$$

La clase *Base* proporciona los métodos para el cambio de base:

ChangeBase	Cambia de base un vector.

La clase *ChangeBasePreprocessor* tiene el siguiente método:

Preprocess	Rota el esqueleto.

7.1.6. Reconocedor de Gestos - *GestureRecognizer*

El reconocedor de gestos calcula el coste de alinear el gesto que queremos detectar con cada gesto entrenado. Este coste se calcula mediante el algoritmo DTW. El gesto reconocido es aquel cuyo coste de alineamiento sea menor.

En la configuración se puede configurar un umbral mínimo entre el primer y el segundo gesto para que el gesto detectado sea válido. En caso de que esté definido dicho umbral, se calcula la diferencia entre el coste de alinear los 2 gestos cuyo coste de alineamiento es menor, y si supera el umbral, se devuelve el gesto.

La clase *GestureRecognizer* contiene las siguientes propiedades:

conf	Configuración de detección.
ctx	Conector con la base de datos.
trainingGestures	Array de gestos entrenados.

La clase *GestureRecognizer* contiene los siguientes métodos:

Evaluate	Realiza la evaluación almacenando los resultados en la base de datos.
GetDistances	Obtiene a lista de costes de alineamiento entre el gesto y cada gesto entrenado.
Recognize	Obtiene y devuelve el gesto detectado.

7.1.7. Extensiones

Las extensiones son clases que sirven de ayuda a otras clases. Hay 3 extensiones de clases:

Estado de un gesto - *State*

El enumerado *State* representa los estados en que puede estar un gesto debido al movimiento:

- Reposo - *Resting*
- Inicio de Movimiento - *StartMoving*
- Movimiento - *Moving*
- Fin de Movimiento - *StopMoving*
- Movimiento terminado - *Finish*

Extensiones del Esqueleto - *SkeletonExtension*

La clase *SkeletonExtension* define los siguientes métodos:

- Center - Centra el esqueleto
- Translate - Traslada el esqueleto dado un vector de traslación
- SkeletonPointToVector3D - Convierte un Skeleton a un Vector3D
- Vector3DToSkeletonPoint - Convierte un Vector3D a un Skeleton
- VectorBetween - Obtiene el vector entre dos puntos
- GetShortestRotationBetweenVectors - Obtiene el cuaternión de rotación entre 2 vectores
- CopySkeleton - Duplica un objeto Skeleton

7.2. Evaluador de Gestos - (*GestureEvaluator*)

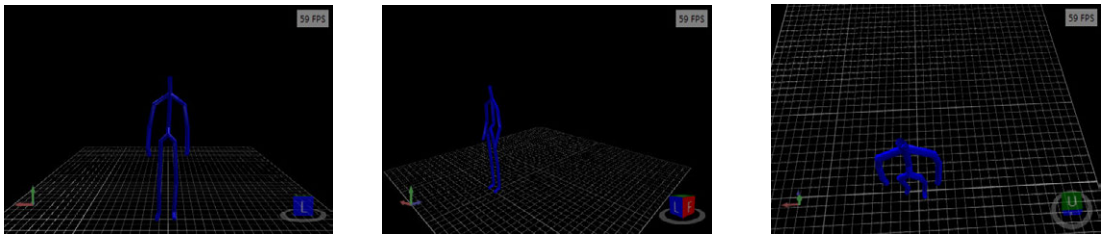
El evaluador de gestos es el proyecto que contiene la parte gráfica (ventanas y páginas) de la aplicación, y desde el cual se hace uso de la librería *Algorithms* para la detección de gestos.

La interfaz gráfica consta de una ventana de navegación que contiene una página. La navegación entre las distintas páginas se realiza con los botones disponibles en las propias páginas. Cuando hacemos click en un botón, se carga la página correspondiente en la ventana de navegación. Como la aplicación es multilenguaje, al cargar la página también se carga el diccionario correspondiente al idioma seleccionado.

```
private void Evaluation_Click(object sender, RoutedEventArgs e)
{
                Page evaluation = new DetectionPage();
                evaluation.Resources.MergedDictionaries.Clear();
                evaluation.Resources.MergedDictionaries.Add(this.Resources.MergedDictionaries.First());
                NavigationService navService = NavigationService.GetNavigationService(this);
                navService.Navigate(evaluation);
}
```

Como vemos en este ejemplo, primero se crea la página DetectionPage. A continuación, se borra el diccionario que hubiera por defecto en este página creada, y se añade el diccionario correspondiente al idioma seleccionado. Y finalmente se navega a la página creada.

En las páginas de reproducción o grabación de gestos, hay un visor donde se muestra el esqueleto en 3 dimensiones y es posible verlo desde cualquier perspectiva en 3 dimensiones: rotando, acercando, alejando...



(a) Vista de frente (b) Vista rotada y alejada (c) Vista desde arriba

Figura 7.8: Visor de gestos en 3 dimensiones

7.2.1. Ventanas

Ventana Inicial - (*MainWindow*)

La ventana inicial es la pantalla que nos aparece nada más ejecutar la aplicación. Podemos acceder introduciendo usuario y contraseña, o podemos crear un nuevo usuario. También es posible cambiar el idioma de la aplicación entre español e inglés.

Figura 7.9: Pantalla de Inicio del Evaluador de Gestos

La clase correspondiente de la ventana inicial es la clase *MainWindow*. Al crearse la clase, se comprueba en un thread aparte que hay comunicación con la base de datos y que la estructura de la misma corresponde con la estructura de la base de datos definida en la aplicación.

La razón de ejecutar la comprobación en un thread aparte es que la comprobación de la comunicación con la base de datos tarda varios segundos, y si no se realiza en un thread aparte, la ventana gráfica no aparecería hasta terminar la comprobación de la base de datos. De esta manera, la ventana gráfica aparece al instante.

Si existe algún problema con la base de datos, bien porque hay un error de comunicación, de credenciales o el modelo de la base de datos no concuerda

con el modelo actual (se ha modificado alguna tabla), la aplicación muestra un mensaje de error y no se puede acceder a la aplicación hasta que se solucione el problema.

La clase *MainWindow* contiene las siguientes propiedades:

bw	BackgroundWorker. Ejecuta procesos de fondo en otro thread.
ctx	Conexión con la base de datos.

La clase *MainWindow* contiene los siguientes métodos:

Button_Click	Comprueba los credenciales del usuario, y si son correctos, va a la Home.
checkDatabase	Comprueba si hay conexión a la base de datos. Si no hay conexión, muestra un mensaje en pantalla.
EnglishClick	Cambia el idioma a inglés
Exit_Click	Se sale de la aplicación
isDtabaseUp	Devuelve true si hay conexión a la base de datos.
MainWindow	Constructor
NewUserClick	Se va a la pantalla de crear un nuevo usuario.
SetLanguageDictionary	Cambia al idioma especificado en el argumento.
SpanishClick	Cambia el idioma a español

Registro de Usuarios - (*RegisterWindow*)

El reconocimiento de gestos es un proceso que requiere de un entrenamiento de gestos previo. Aunque teóricamente se podría usar un sólo entrenamiento para reconocer los gestos realizados por cualquier persona, en la práctica observamos que cada persona no realiza los mismos gestos de la misma manera. Por ejemplo, el gesto de levantar la mano se puede realizar desde una altura hasta otra altura, y a distintas velocidades.

Por este motivo, el evaluador de gestos tiene un sistema de usuarios donde cada usuario realiza su propio entrenamiento y puede realizar sus propias evaluaciones.

El registro de usuario es muy simple, y tiene los campos de Nombre, Contraseña e Idioma:

- El **Nombre** es un elemento único, por lo que no puede haber 2 usuarios con el mismo nombre. Es similar al Login de un sistema operativo, correo o cualquier aplicación, que es único.
- La **Contraseña** puede ser cualquier combinación de caracteres. Por motivos de seguridad, en la base de datos se guarda simplemente el hash de la contraseña.
- El **Idioma** permite mostrar la aplicación para el usuario por defecto en Inglés o en Español. Realizar una traducción a otro idioma es muy sencillo, ya que sólo hay que traducir el fichero StringResources.xaml.

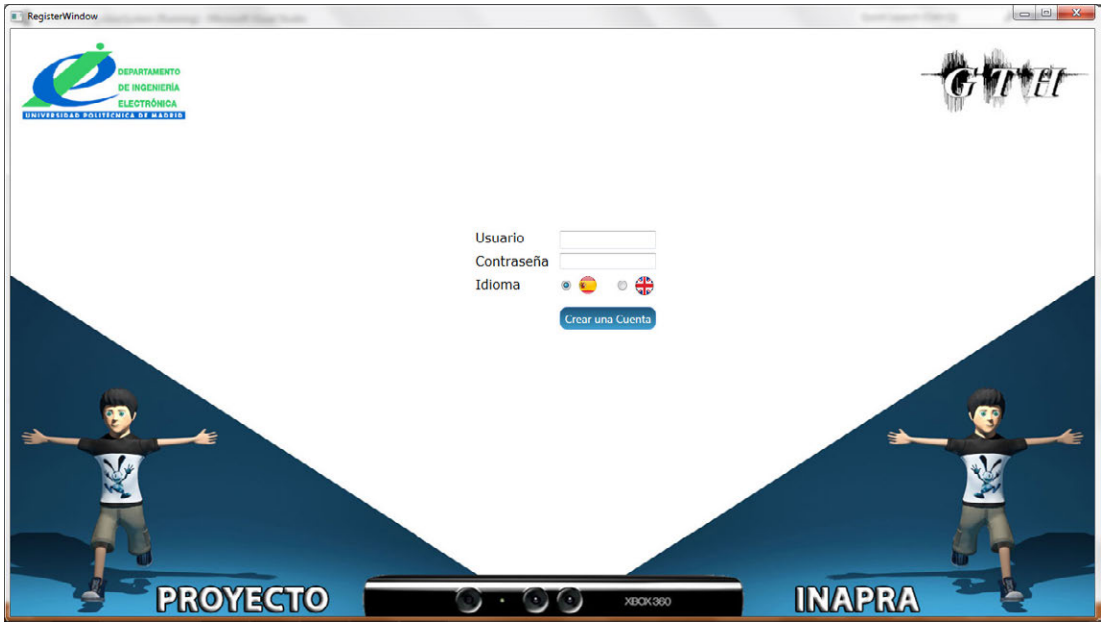


Figura 7.10: Pantalla de Registro de Usuarios

La clase *RegisterWindow* contiene las siguientes propiedades:

ctx	Conexión con la base de datos.

La clase *RegisterWindow* contiene los siguientes métodos:

Register_Click	Crea un nuevo Usuario en la base de datos con los campos introducidos: usuario, contraseña e idioma. Si algún campo está vacío, se muestra un error en pantalla y no se crea el usuario en la base de datos.
RegisterWindow	Constructor.

Ventana de navegación Principal - (*MainWindow*)

La clase *HomeWindow* es una ventana de navegación (*NavigationWindow*) que hace de contenedor para las distintas páginas. En esta ventana se irán cargando las distintas páginas, incluida la propia página del Home (*HomePage*).

La clase *HomeWindow* contiene las siguientes propiedades:

username	Nombre del usuario que se ha logueado. Esta propiedad es una propiedad estática y es utilizado en los accesos a base de datos para especificar el usuario. En una página web sería el equivalente a una variable de sesión.

La clase *HomeWindow* contiene los siguientes métodos:

HomeWindow	Constructor.

7.2.2. Páginas

Página principal - (*HomePage*)

Una vez introducidos el usuario y contraseña, se accede a la página principal.

Desde esta pantalla se puede ir a *Entrenamiento* para realizar el entrenamiento de gestos, o *Detección*, para realizar una evaluación o detectar gestos continuamente.

Desde la página principal se pueden realizar las siguientes acciones:

- Ir a la página de Entrenamiento
- Ir a la página de Detección
- Cerrar Sesión, haciendo click en el botón *Cerrar Sesión*
- Salir de la aplicación, haciendo click en el botón *Salir*
- Cambiar de idioma, haciendo click en el botón con la bandera español (Español) o en el botón con la bandera inglesa (Inglés).

Figura 7.11: Home del Evaluador de Gestos

La clase *HomePage* contiene las siguientes propiedades:

ctx	Conexión con la base de datos.

La clase *HomePage* contiene los siguientes métodos:

CloseSession_Click	Cierra la sesión. La aplicación va a la pantalla principal.
English_Click	Cambia el idioma a inglés
Evaluation_Click	Se va a la página de Evaluación.
Exit_Click	Se sale de la aplicación.
HomePage	Constructor.
SetLanguageDictionary	Cambia al idioma especificado en el argumento.
Spanish_Click	Cambia el idioma a español
Training_Click	Se va a la página de Entrenamiento.

Entrenamiento - (*TrainingHomePage*)

Desde la página de Entrenamiento podemos acceder a la *Realización del Entrenamiento*, a la *Configuración del Entrenamiento* o a los *Gestos Entrenados*.

Figura 7.12: Entrenamiento

La clase *TrainingHomePage* contiene los siguientes métodos:

Configure_Click	Accede a la configuración del entrenamiento.
ModifyGestures_Click	Accede a la página *GesturesPage*.
Start_Click	Empieza el entrenamiento.

Configuración - (*ConfigurationPage*)

La configuración dentro del apartado de Entrenamiento tiene 2 opciones:

- Configuración de Detección de Gestos
- Configuración del Entrenamiento

Desde la Configuración de *Detección de Gestos*, se puede ir a la página *RecognitionConfigurationPage* que permite seleccionar el perfil de Configuración para la detección de gestos. Por ejemplo, si seleccionamos el perfil de Esqueleto Centrado y Rotado, estaremos especificando que para detectar un gesto sólo se consideren hombros, codos, muñecas y manos y que el esqueleto esté centrado y rotado para que mire siempre de frente. Con esta configuración nos podríamos mover por el escenario sin que se detectara como un gesto. El movimiento del tronco, piernas y pies no se tiene en cuenta, y el correspondiente al desplazamiento de hombros, codos, muñecas y manos tampoco al estar el esqueleto siempre centrado y de frente.

Desde la Configuración del *Entrenamiento*, se puede ir a la página *ConfigurationTrainingPage* donde se pueden configurar los gestos a realizar en el entrenamiento.

Figura 7.13: Configuracion

Configuración de Entrenamiento - (*ConfigurationTrainingPage*)

La configuración del entrenamiento permite seleccionar los distintos gestos que definen el entrenamiento. Los gestos entrenados serán los utilizados posteriormente como patrón a la hora de detectar un gesto.

Un gesto del entrenamiento puede estar habilitado o deshabilitado:

- Los gestos **habilitados** son los que aparecen en el entrenamiento y se utilizan como referencia a la hora de detectar un gesto.
- Los gestos **deshabilitados** no se utilizan.

Además de la configuración de los gestos del entrenamiento, esta página permite configurar el entrenamiento en **modo automático**:

- Un entrenamiento **automático** reproduce y graba los gestos del entrenamiento continuamente, sin interrupciones.
- Si está en modo **no automático**, después de cada gesto grabado del entrenamiento el sistema permite reproducir el gesto grabado, volver a grabar el gesto o continuar el entrenamiento con el siguiente gesto.

Figura 7.14: Configuracion de Entrenamiento

La clase *ConfigurationTrainingPage* contiene las siguientes propiedades:

ctx	Conector con la base de datos.
gesturesDisabled	Gestos del entrenamiento deshabilitados.
gesturesEnabled	Gestos del entrenamiento habilitados.
imagePath	Ruta de la imagen de un gesto de entrenamiento (imagen que se reproduce en el entrenamiento para guiar al usuario en la realización del gesto).

La clase *ConfigurationTrainingPage* contiene los siguientes métodos:

Add_Click	Habilita un gesto del entrenamiento.
AddGesture_Click	Crea un nuevo gesto.
Delete_Click	Deshabilita un gesto del entrenamiento.
GestureImage_Click	Añada una imagen a un gesto del entrenamiento.
Save_Click	Guarda la configuración.
UpdateList	Actualiza la tabla de gestos habilitados y deshabilitados.

Comenzar Entrenamiento - (*StartTrainingPage*)

El entrenamiento consiste en la grabación de los gestos configurados en el apartado de Configuración. Para ello se sigue el siguiente proceso:

1. Se muestra el nombre del gesto con el número de gesto del entrenamiento.

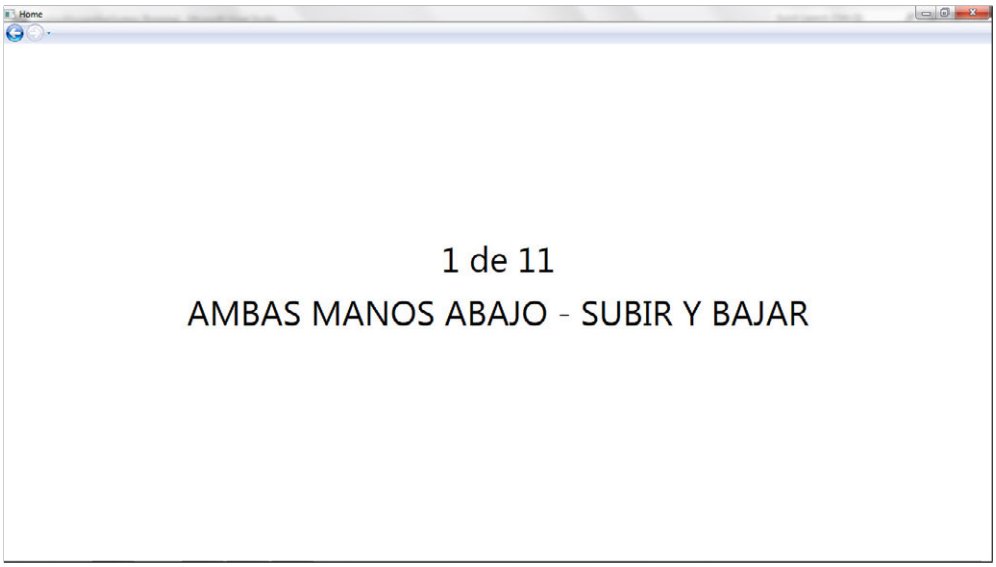


Figura 7.15: Inicio de la animación del Gesto

2. A continuación se muestra una animación del gesto a entrenar.

(a) Animacion 1 (b) Animacion 2 (c) Animacion 3 (d) Animacion 4

Figura 7.16: Animacion del Gesto Levantar Ambas Manos

3. Se espera 2 segundos después de reproducir la animación para que el usuario se prepare para realizar el gesto.

4. Se muestra la pantalla con el esqueleto detectado y se graba el gesto.

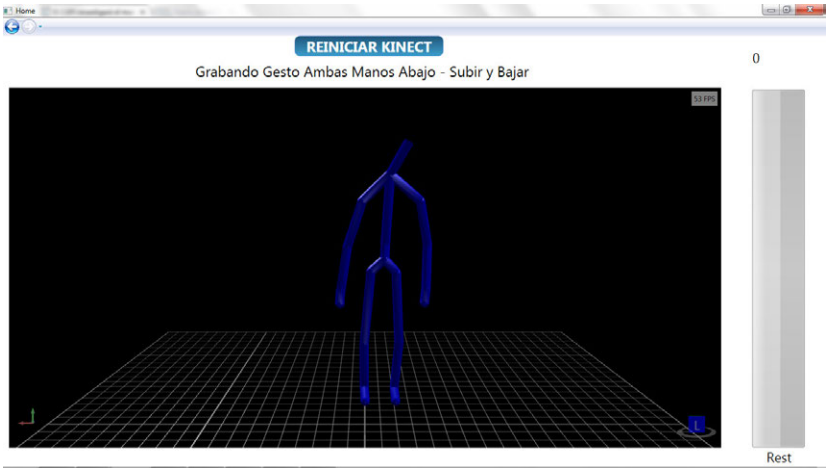


Figura 7.17: Inicio de la Grabación del Gesto

5. Si el entrenamiento es automático, se vuelve al punto 1) con el siguiente gesto. Si no es automático, se muestra la siguiente pantalla.

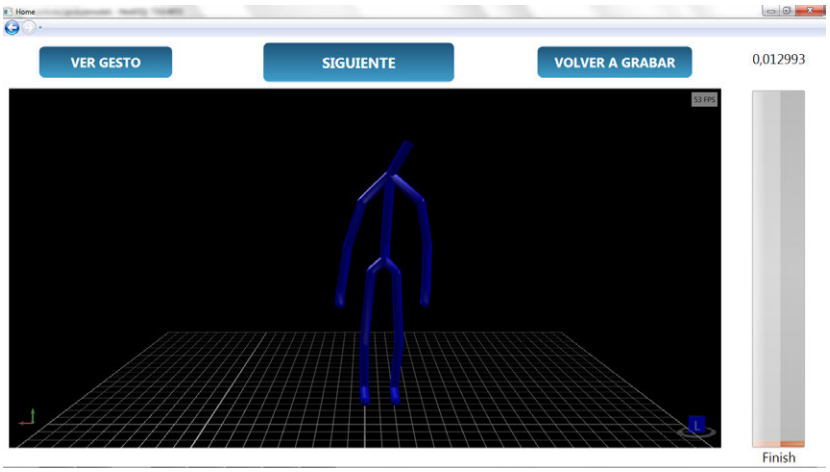


Figura 7.18: Fin de la Grabación del Gesto

El gesto que se graba corresponde al movimiento completo detectado por la máquina de estados. El sensor Kinect está continuamente activo, y la máquina de estados empieza a procesar los frames una vez que la variable *enableRecord* está activa. Esta variable está activa desde que termina la animación del gesto del entrenamiento y termina la grabación del gesto actual. El gesto comienza a grabarse desde que la máquina de estados pasa del estado *Reposo* a *Inicio de Movimiento*; y se termina de grabar cuando pasa del estado *Fin de Movimiento* a *Reposo*.

El Kinect está continuamente activo porque cuando se activa, tiene detectar el esqueleto. Una vez detectado, Kinect simplemente realiza un tracking del esqueleto. La detección del esqueleto es un proceso complejo que lleva cierto tiempo. A veces es rápido, pero otras veces el proceso puede tardar varios segundos. Sin embargo el proceso de tracking es un proceso sencillo e inmediato, con lo que nos garantiza que una vez detectado el esqueleto a principio del entrenamiento, Kinect va a seguir el esqueleto durante todo el entrenamiento.

La clase *StartTrainingPage* contiene las siguientes propiedades:

automatic	Indica si el entrenamiento es automático.
counter	Contador de gestos: número de gesto grabados + 1.
ctx	Conector con la base de datos.
currentGesture	Gesto del entrenamiento que se está grabando.
enableRecord	Indica si se pueden capturar los frames.
images	Lista de imágenes del entrenamiento.
numGestures	Número total de gestos del entrenamiento.
sensor	Conector con el Kinect.
trainingGestures	Lista de gestos del entrenamiento.

La clase *StartTrainingPage* contiene los siguientes métodos:

ButtonFinish_Click	Termina el entrenamiento.
finishGesture	Acción cuando termina el gesto.
KinectRestart	Reinicia el Kinect.
LoadGif	Reproduce una imagen gif.
NextGesture	Empieza el entrenamiento del siguiente gesto.
Next_Click	Si el modo automático está deshabilitado, pasa al siguiente gesto.
ReplayGif_Click	Repite la animación del gesto entrenado.
Replay_Click	Si el modo automático está deshabilitado, repite el gesto grado.
ReRecord_Click	Si el modo automático está deshabilitado, vuelve a grabar el gesto.
sensor_SkeletonFrameReadyRecord	Graba cada frame en el gesto.
sensor_SkeletonFrameReadyReplay	Reproduce cada frame del gesto grabado.

Gestos Grabados - (*GesturesPage*)

Desde la página de Gestos Grabados (*GesturesPage*), se puede:

- Ver gestos del entrenamiento grabados (VER GESTOS)
- Grabar gestos del entrenamiento (AÑADIR GESTOS)
- Borrar gestos del entrenamiento (BORRAR GESTOS)
- Modificar gestos del entrenamiento (MODIFICAR GESTOS)

Figura 7.19: Página de Gestos del Entrenamiento

La clase *GesturesPage* contiene los siguientes métodos:

Add_Click	Accede a la página de *Grabar gestos del entrenamiento*
Delete_Click	Accede a la página de *Borrar gestos del entrenamiento*
Modify_Click	Accede a la página de *Modificar gestos del entrenamiento*
Play_Click	Accede a la página de *Reproducir gestos del entrenamiento*

Añadir Gestos - (*AddGesturesPage*)

Desde esta página se pueden añadir y grabar gestos de entrenamiento. El proceso es sencillo: se introduce el nombre del gesto de entrenamiento y se da al botón de grabar.

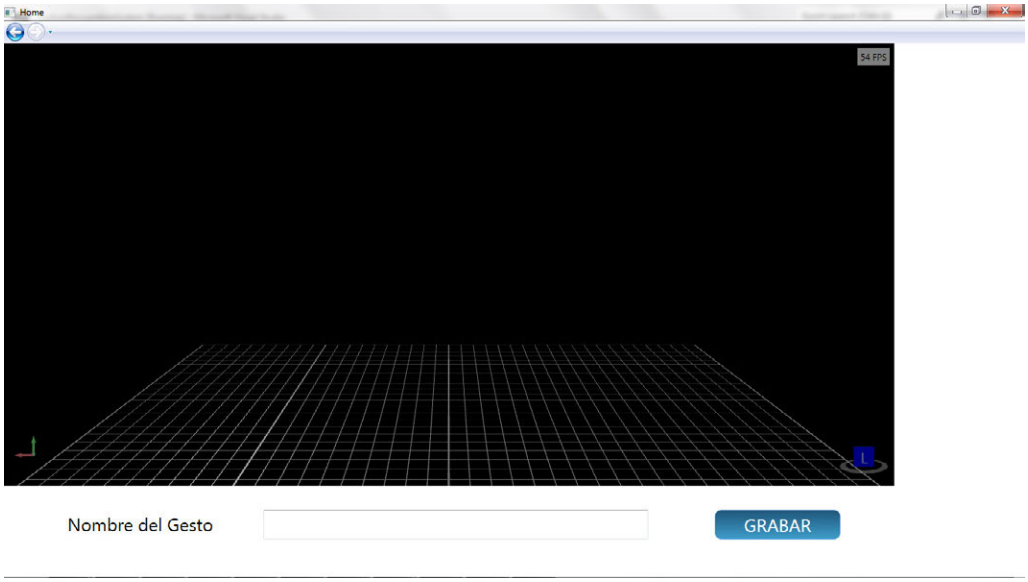


Figura 7.20: Página de Añadir Gesto

La clase *AddGesturesPage* contiene las siguientes propiedades:

ctx	Conector con la base de datos.
gesture	Gesto para añadir.
sensor	Conector con el Kinect.

La clase *AddGesturesPage* contiene los siguientes métodos:

finishGesture	Cuando finaliza el gesto, se llama a este método.
NavigationService_Navigating	Desactiva el kinect y vuelve atrás en el menú de navegación.
Record_Click	Comienza la grabación del gesto.
RepeatGesture_Click	Reproduce el gesto grabado.
ReRecord_Click	Graba el gesto de nuevo.
sensor_SkeletonFrameReadyRecord	Graba cada frame en el gesto.
sensor_SkeletonFrameReadyReplay	Reproduce cada frame del gesto grabado.

Borrar Gestos - (*DeleteGesturesPage*)

Desde esta página se pueden reproducir y borrar gestos grabados en el entrenamiento.

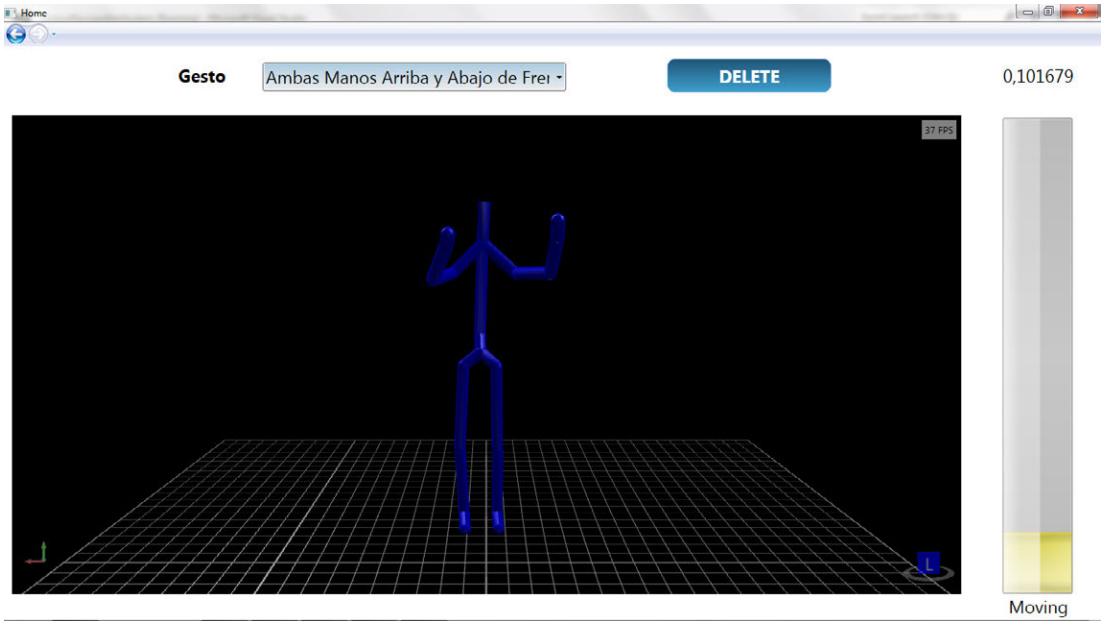


Figura 7.21: Página de Borrar Gestos

La clase *DeleteGesturesPage* contiene las siguientes propiedades:

ctx	Conector con la base de datos.
gesture	Gesto que se va a borrar.

La clase *DeleteGesturesPage* contiene los siguientes métodos:

DeleteGesturesPage_Click	Borra el gesto seleccionado.
GestureList_SelectionChanged	Cuando se selecciona un gesto, lo reproduce para que se pueda comprobar antes de borrar.
sensor_SkeletonFrameReadyReplay	Reproduce cada frame del gesto grabado.

Modificar Gestos - (*ModifyGesturesPage*)

Desde esta página se pueden reproducir y modificar gestos grabados en el entrenamiento.

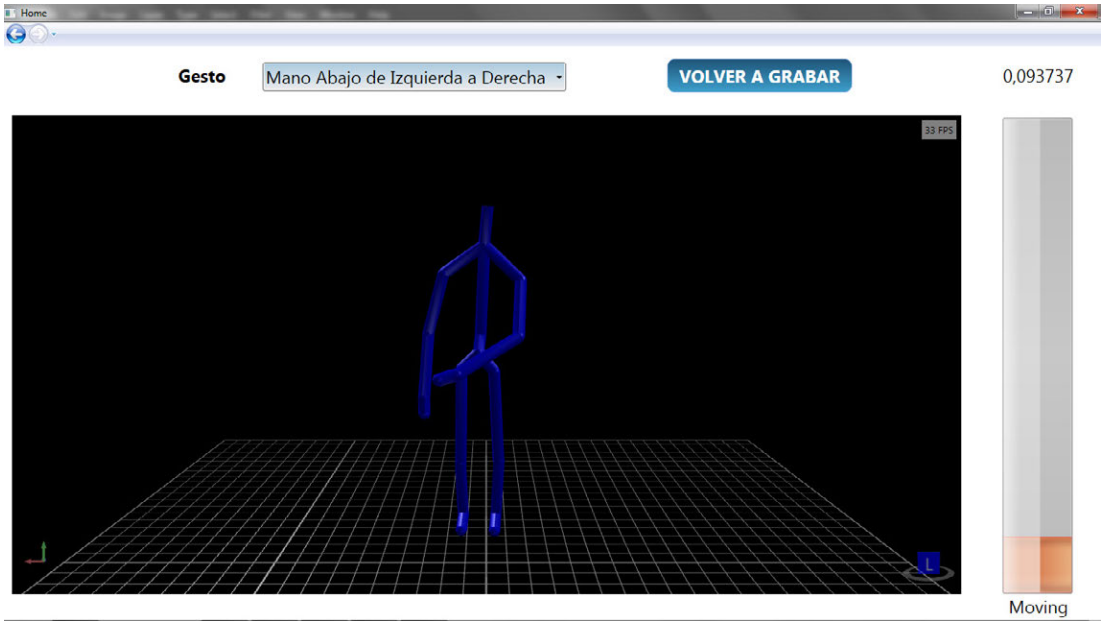


Figura 7.22: Página para Modificar Gestos

La clase *ModifyGesturesPage* contiene las siguientes propiedades:

ctx	Conector con la base de datos.
gesture	Gesto que se va a borrar.

La clase *ModifyGesturesPage* contiene los siguientes métodos:

GestureList_SelectionChanged	Cuando se selecciona un gesto, lo reproduce para que se pueda comprobar antes de borrar.
ModifyGesturesPage_Click	Modifica el gesto seleccionado.
ReRecord_Click	Vuelve a grabar el gesto seleccionado.
sensor_SkeletonFrameReadyReplay	Reproduce cada frame del gesto grabado.

Ver Gestos - (*PlayGesturePage*)

Desde esta página se pueden reproducir los gestos grabados en el entrenamiento.

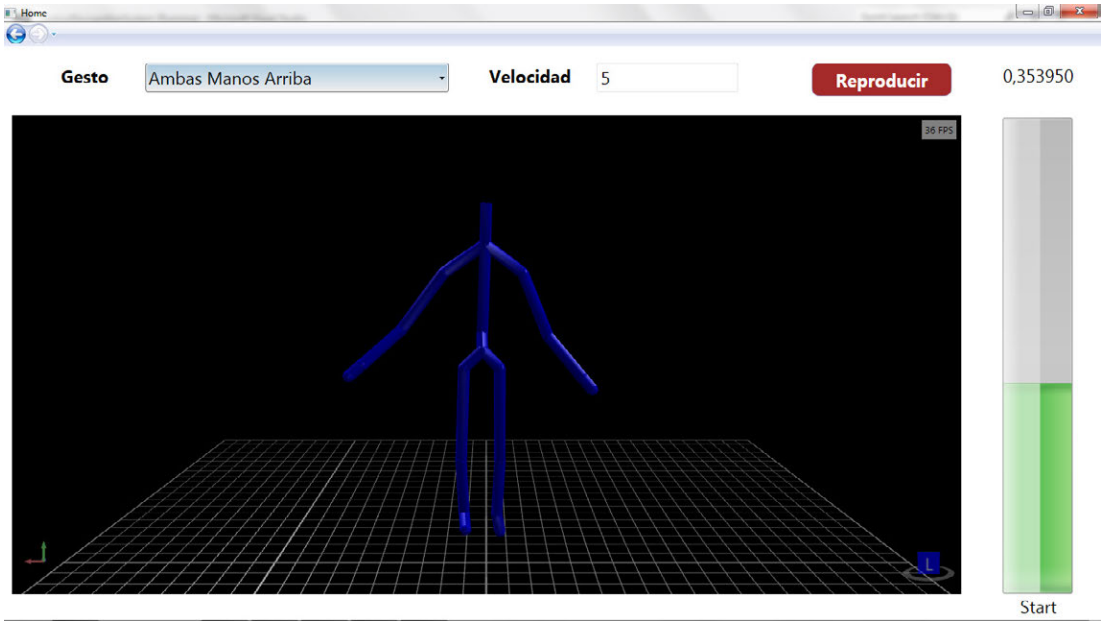


Figura 7.23: Página para Reproducir Gestos

La clase *PlayGesturePage* contiene las siguientes propiedades:

ctx	Conector con la base de datos.
gesture	Gesto que se va a borrar.

La clase *PlayGesturePage* contiene los siguientes métodos:

Play_Click	Reproduce el gesto seleccionado.
sensor_SkeletonFrameReadyReplay	Reproduce cada frame del gesto grabado.

Detección - (*DetectionPage*)

Desde la página de *Detección* podemos acceder a la *Detección en Modo Libre*, a la *Configuración de la Detección* o a realizar una *Evaluación*.

Figura 7.24: Detección

La clase *DetectionPage* contiene los siguientes métodos:

Configuration_Click	Accede a la configuración de la detección.
Evaluation_Click	Accede a la página *EvaluationHomePage*.
FreeMode_Click	Empieza la detección de gestos de forma continua.

Modo Libre - (*FreeModePage*)

La detección de gestos en Modo Libre está continuamente detectando gestos. Cuando detecta un gesto, muestra un mensaje del gesto detectado.

La clase *FreeModePage* contiene las siguientes propiedades:

bw	BackgroundWorker. Ejecuta procesos de fondo en otro thread.
ctx	Conector con la base de datos.
sensor	Conector con el Kinect.
recognizer	Reconocedor de gestos.

gesture	Gesto grabado.
detected	Gesto detectado.
automatic	Indica si la detección es automática.
enableRecord	Indica si puede grabar el gesto.

La clase *FreeModePage* contiene los siguientes métodos:

Detect_Click	
Replay_Click	Repite el grabado grabado.
startGestureRecognizer	Proceso para reconocer el gesto grabado.
finishGesture	Método cuando termina el gesto.
sensor_SkeletonFrameReadyRecord	Graba cada frame que envía el Kinect.
sensor_SkeletonFrameReadyRecordedReplay	Reproduce cada frame del gesto grabado.
sensor_SkeletonFrameReadyDetectedReplay	Reproduce cada frame del gesto detectado.
NavigationService_Navigating	Desactiva el kinect y vuelve atrás en el menú de navegación.

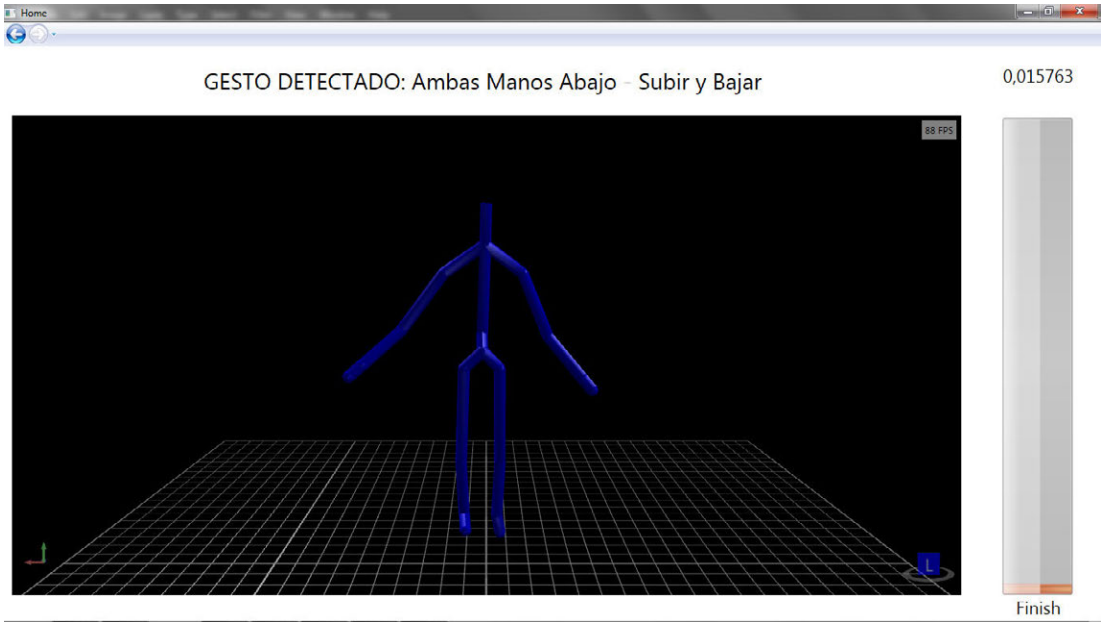


Figura 7.25: Detección en Modo Libre

Selección de Evaluación - (*EvaluationHomePage*)

Desde la página de *EvaluationHomePage* podemos:

- *Iniciar la Evaluación de Gestos Aleatorios*
- *Iniciar la Evaluación de Gestos Configurados*
- *Calcular los Resultados de una Evaluación*
- *Ver los Resultados de una Evaluación*

Figura 7.26: Selección de Evaluación

La clase *EvaluationHomePage* contiene los siguientes métodos:

CalculateEvaluationResults	Calcula los Resultados de una Evaluación.
CustomEvaluation_Click	Inicia la Evaluación de Gestos Configurados.
EvaluationResults_Click	Muestra los Resultados de una Evaluación.
RandomEvaluation_Click	Inicia la Evaluación de Gestos Aleatorios.

Evaluación - (*EvaluationPage*)

La evaluación consiste en la grabación de unos gestos que sabemos a priori cuáles son. En la Evaluación de Gestos Configurados, los gestos son seleccionados en el apartado de Configuración. En la Evaluación de Gestos Aleatorios, los gestos se seleccionan aleatoriamente. El proceso de la evaluación es el siguiente:

1. Se muestra el nombre del gesto con el número de gesto sobre el total.

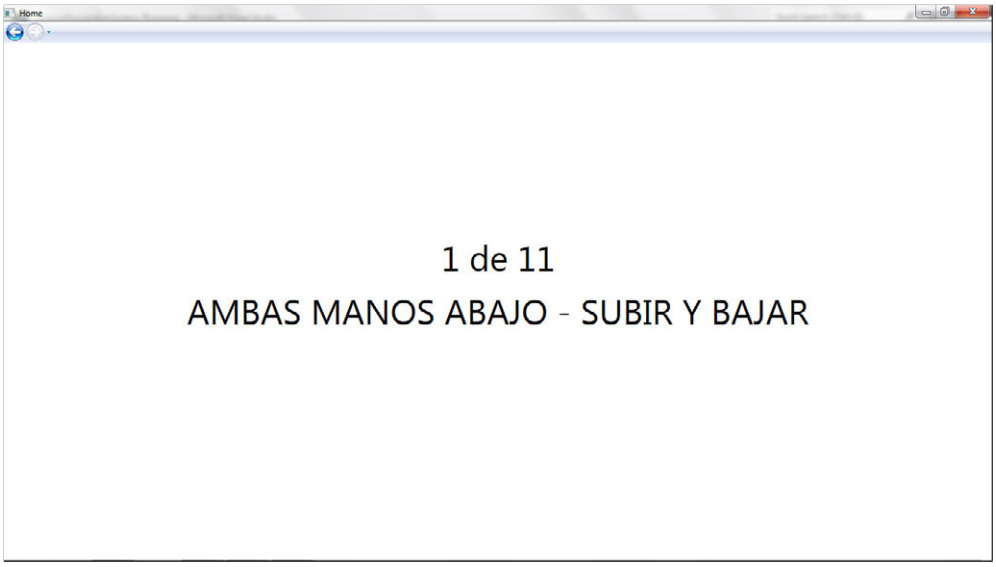


Figura 7.27: Inicio de la animación del Gesto

2. A continuación se muestra una animación del gesto a entrenar.

(a) Animacion 1 (b) Animacion 2 (c) Animacion 3 (d) Animacion 4

Figura 7.28: Animacion del Gesto Levantar Ambas Manos

3. Se espera 2 segundos después de reproducir la animación para que el usuario se prepare para realizar el gesto.

4. Se muestra la pantalla con el esqueleto detectado y se graba el gesto.

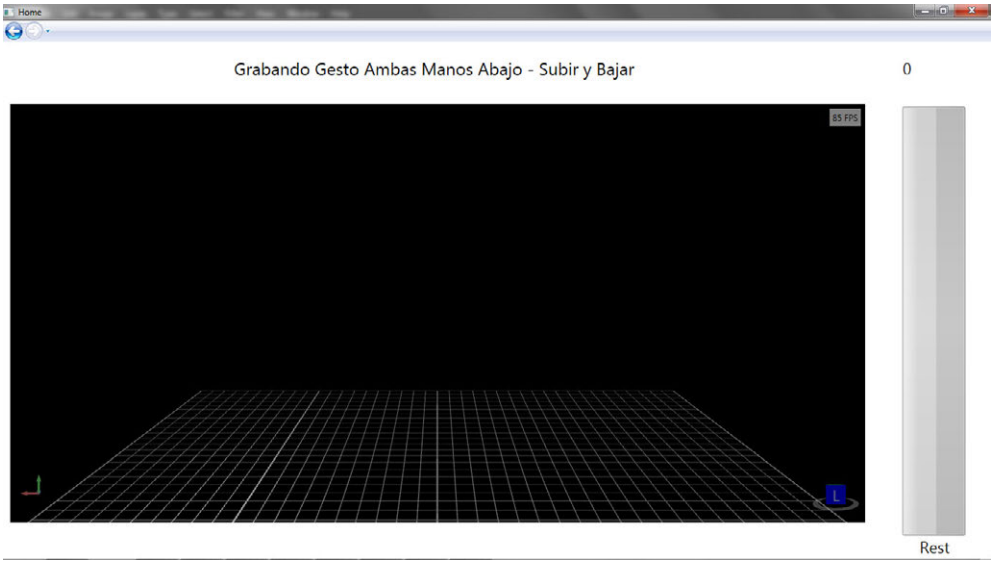


Figura 7.29: Inicio de la Grabación del Gesto de la Evaluación

5. Si la evaluación es automática, se vuelve al punto 1) con el siguiente gesto. Si no es automática, aparecen 3 botones: Reproducir el gesto grabado, Volver a grabar el gesto y Siguiente Gesto de la Evaluación.

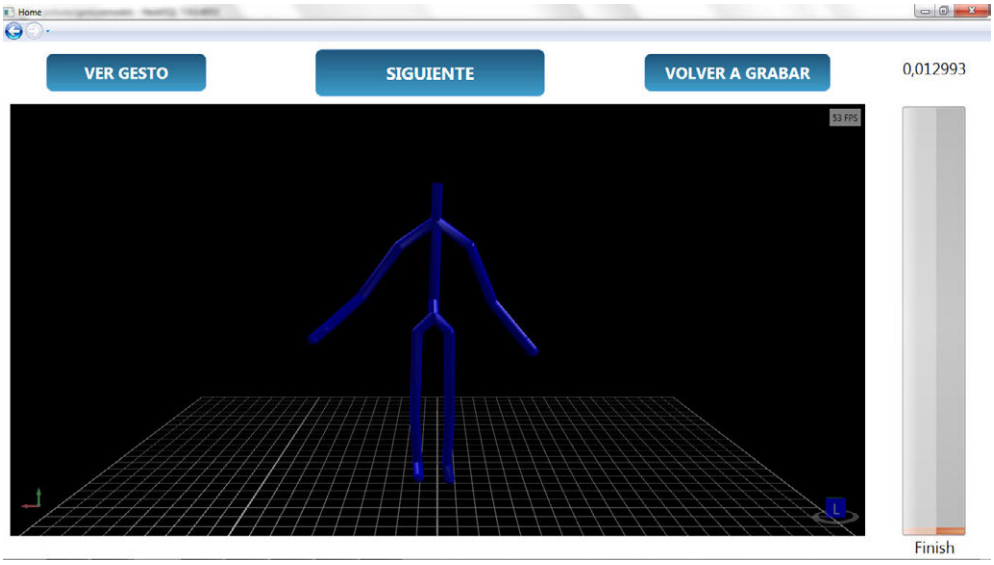


Figura 7.30: Fin de la Grabación del Gesto de la Evaluación

La clase *EvaluationPage* contiene las siguientes propiedades:

automatic	Indica si la evaluación es automática.
counter	Contador de gestos: número de gesto grabados + 1.
ctx	Conector con la base de datos.
currentGesture	Gesto de la evaluación que se está grabando.
currentTrainingGesture	Gesto entrenado que se está evaluando.
enableRecord	Indica si se pueden capturar los frames.
evaluation	Entrada de la base de datos de la Evaluación.
images	Lista de imágenes de los gestos entrenados.
numGestures	Número total de gestos del entrenamiento.
sensor	Conector con el Kinect.
trainingGestures	Lista de gestos entrenados.

La clase *EvaluationPage* contiene los siguientes métodos:

ButtonFinish_Click	Termina la evaluación.
finishGesture	Acción cuando termina el gesto.
LoadGif	Reproduce una imagen gif.
NavigationService_Navigating	Desactiva el kinect y vuelve atrás en el menú de navegación.
Next_Click	Si el modo automático está deshabilitado, pasa al siguiente gesto.
NextGesture	Empieza la evaluación del siguiente gesto.
Replay_Click	Si el modo automático está deshabilitado, repite el gesto grado.
ReplayGif_Click	Repite la animación del gesto evaluado.
ReRecord_Click	Si el modo automático está deshabilitado, vuelve a grabar el gesto.
sensor_SkeletonFrameReadyRecord	Graba cada frame en el gesto.
sensor_SkeletonFrameReadyReplay	Reproduce cada frame del gesto grabado.

Calcular los Resultados de una Evaluación - (*CalculateEvaluationResultsPage*)

Una vez realizada una evaluación, se pueden calcular los resultados de la evaluación. Para evitar tener que esperar al cálculo de los resultados después de realizar la evaluación, se ha separado el cálculo de los resultados de la evaluación en sí. De esta manera es posible realizar varias evaluaciones seguidas sin tener que esperar al cálculo de los resultados.

Figura 7.31: Cálculo de Resultados de la Evaluación

La clase *CalculateEvaluationResultsPage* contiene las siguientes propiedades:

ctx	Conector con la base de datos.
evaluation	Evaluación seleccionada para calcular los resultados.

La clase *CalculateEvaluationResultsPage* contiene los siguientes métodos:

Calculate_Click	Calcula los resultados de la evaluación.
Evaluate	Calcula los resultados de la evaluación.
EvaluateResults	Calcula el resumen de los resultados de la evaluación.

Ver los Resultados de una Evaluación - (*EvaluationResultPage*)

Una vez calculados los resultados de una evaluación, se pueden consultar los resultados. Para consultar los resultados de una evaluación, primero es necesario seleccionar la evaluación que queremos consultar (tenemos que haber calculado previamente los resultados de la misma).

Figura 7.32: Selección de la evaluación para ver los resultados

Una vez seleccionada una evaluación, nos aparecerá una pantalla con los resultados en 3 ventanas en la parte superior, y en la parte inferior tendremos otras 3 ventanas de reproducción de gestos.

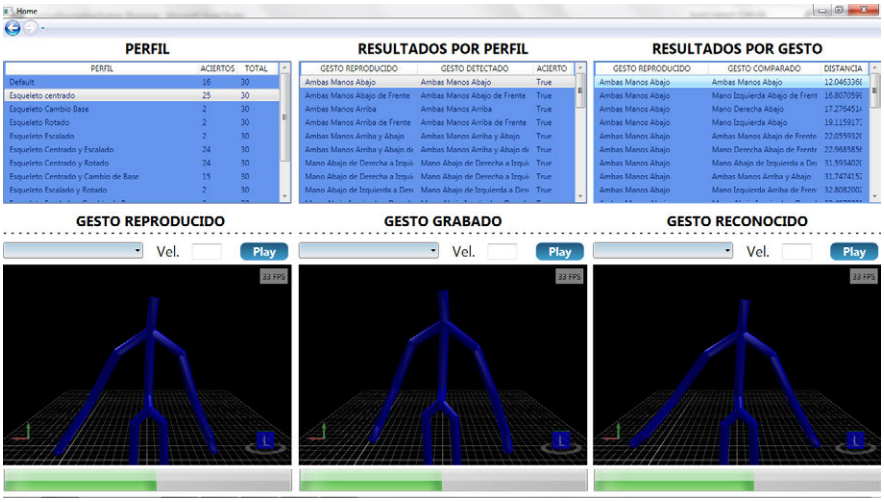


Figura 7.33: Pantalla de Consulta de Resultados de la Evaluación

En la pantalla de resultados, a la izquierda del todo tenemos un listado con todos los perfiles de configuración de la evaluación, con el número de gestos acertados y el número de gestos totales de la evaluación. Si seleccionamos un perfil de configuración, nos aparecerá en la ventada de en medio los resultados para esa evaluación.

Estos resultados se muestran con el siguiente formato: gesto mostrado en la evaluación a la izquierda, gesto detectado en medio, y resultado (acierto o fallo) en la derecha. Si seleccionamos un resultado, podremos ver en la ventana de la derecha todos los resultados de aplicar el algoritmo DTW al gesto de la evaluación.

Los resultados de aplicar el algoritmo DTW incluyen las distancias de cada gesto grabado con cada gesto entrenado usando todos el perfil de configuración seleccionado.

Figura 7.34: Pantalla de Consulta de Resultados de la Evaluación: Vista Superior

En la parte inferior, la primera ventana muestra el gesto entrenado seleccionado que nos apareció al realizar la evaluación. La segunda ventana muestra el gesto grabado en la evaluación. Y la tercera ventana, muestra el gesto detectado. En el caso de que la detección haya sido correcta para el gesto de la evaluación seleccionada, el gesto entrenado y el gesto detectado serán los mismos. Los gestos se pueden reproducir a cualquier velocidad, así como verlos en 3 dimensiones: girar la vista, acercarse, alejarse...

Figura 7.35: Pantalla de Consulta de Resultados de la Evaluación: Vista Inferior

La clase *EvaluationResultPage* contiene las siguientes propiedades:

ctx	Conecto con la base de datos.
evaluation	Evaluación seleccionada.
evaluationItemResults	Elementos de la evaluación seleccionada.
evaluationResults	Resultados de la evaluación seleccionada.
evaluationResultsDTW	Resultados DTW de la evaluación seleccionada.
gestureDetected	Gesto detectado seleccionado.
gestureRecorded	Gesto grabado seleccionado.
gestureReplayed	Gesto entrenado seleccionado.
preprocessor	Preprocesamiento correspondiente al perfil de configuración seleccionado.
user	Usuario al que pertenece la evaluación.

La clase *EvaluationResultPage* contiene los siguientes métodos:

Initialize	Inicializa la vista de configuraciones y resultados de la evaluación seleccionada.
PlayGestureDetected	Reproduce el gesto detectado seleccionado.
PlayGestureRecorded	Reproduce el gesto grabado seleccionado.
PlayGestureReplayed	Reproduce el gesto entrenado seleccionado.
PlayGestures	Reproduce los gestos entrenado, grabado y detectado.
GestureDetectedPlay_Click	Reproduce los gestos entrenado, grabado y detectado seleccionados.
GestureRecordedPlay_Click	Reproduce los gestos entrenado, grabado y detectado seleccionados.
GestureReplayedPlay_Click	Reproduce los gestos entrenado, grabado y detectado seleccionados.
ProfileList_SelectionChanged	Método que se invoca cuando se selecciona perfil de configuración.
DTWResultsList_SelectionChanged	Método que se invoca cuando se selecciona un elemento de la lista de resultados DTW.

EvaluationListView_SelectionChanged	Método que se invoca cuando se selecciona un elemento de la lista de evaluaciones.
EvaluationResultsList_SelectionChanged	Método que se invoca cuando se selecciona un elemento de la lista de resultados de la evaluación.
GestureDetectedCombobox_SelectionChanged	Método que se invoca cuando se selecciona un elemento de la lista de gestos detectados.
GestureRecordedCombobox_SelectionChanged	Método que se invoca cuando se selecciona un elemento de la lista de los gestos grabados.
GestureReplayedCombobox_SelectionChanged	Método que se invoca cuando se selecciona un elemento de la lista de los gestos entrenados.
sensor_SkeletonFrameReadyGestureDetected	Reproduce los frames del gesto detectado.
sensor_SkeletonFrameReadyGestureRecorded	Reproduce los frames del gesto grabado.
sensor_SkeletonFrameReadyGestureReplayed	Reproduce los frames del gesto entrenado.

Configuración - (*EvaluationConfigurationPage*)

Desde la Configuración de *Detección* se puede acceder a 4 configuraciones: **Detección de Gestos**, **Modo Libre**, **Evaluación de Gestos Aleatorios**, **Evaluación de Gestos Aleatorios**

Figura 7.36: Configuracion Detección

La clase *EvaluationConfigurationPage* contiene los siguientes métodos:

CustomEvaluationConfiguration	Accede a la configuración de la Evaluación de Gestos Configurados.
FreeModeConfiguration_Click	Accede a la configuración de la detección continua (Mode Libre).
RandomConfiguration_Click	Accede a la configuración de la Evaluación de Gestos Aleatorios.
RecognitionConfiguration_Click	Accede a la configuración de la Detección de Gestos.

Configuración de la Detección en Modo Libre - (*FreeModeConfigurationPage*)

En la Configuración del *Modo Libre* se pueden configurar la detección automática y el perfil de configuración de detección.

Figura 7.37: Configuracion Detección en Modo Libre

Si se habilita la detección automática, el sistema está detectando gestos continuamente. Por el contrario, si se dehabilita la detección automática, el sistema pare después de la detección de cada gesto, lo que permite reproducir el gesto grabado.

El perfil de Configuración para la detección de gestos permite seleccionar la configuración que queremos que se aplique para detectar gestos. Por ejemplo, si seleccionamos el perfil de Esqueleto Centrado, estaremos especificando que para detectar un gesto sólo se consideren hombros, codos, muñecas y manos y que el esqueleto esté siempre centrado. Con esta configuración nos podríamos mover por el escenario sin que se detectara como un gesto. El movimiento del tronco, piernas y pies no se tiene en cuenta, y el correspondiente al desplazamiento de hombros, codos, muñecas y manos tampoco al estar el esqueleto siempre centrado. Con los giros tendríamos problemas, por lo que si hubiera giros sería recomendable utilizar el perfil de Centrado y Rotado.

Configuración de la Evaluación de Gestos Aleatorios - (*RandomEvaluationConfigurationPage*)

En la Configuración de la *Evaluación de Gestos Aleatorios* se puede configurar si la evaluación es automática o no y el perfil de configuración de detección.

Si se habilita la detección automática, en la evaluación se graban gestos continuamente. Si no se habilita la detección automática, el sistema para después de la grabación de cada gesto, lo que permite reproducir o volver a grabar un gesto.

El perfil de Configuración para la detección de gestos permite seleccionar la configuración que queremos que se aplique para detectar gestos.

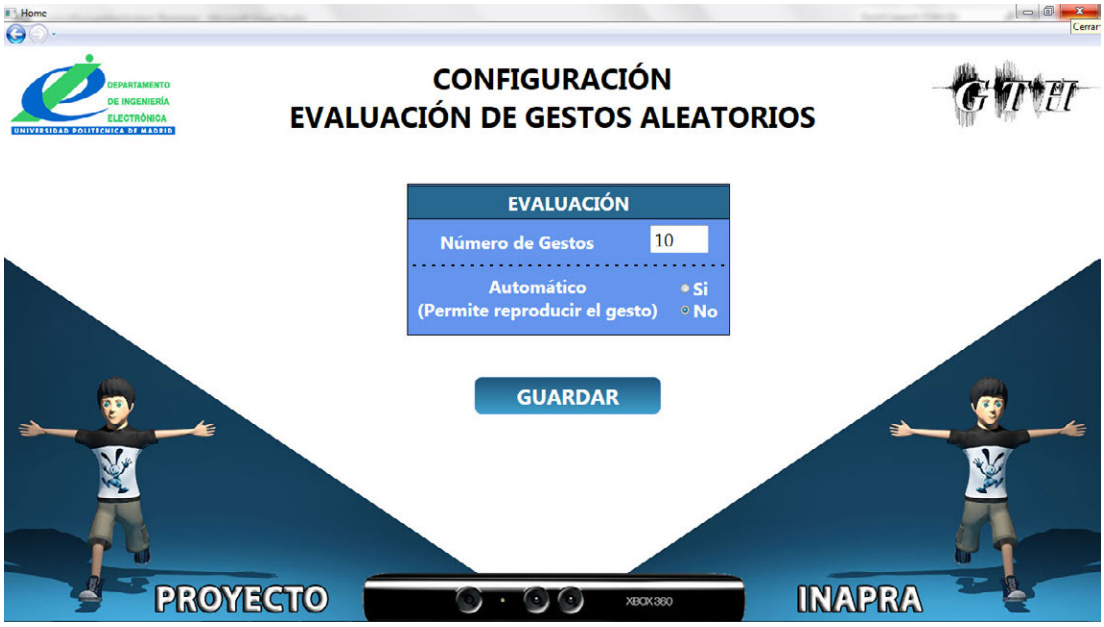


Figura 7.38: Configuración de la Evaluación de Gestos Aleatorios

La clase *RandomEvaluationConfigurationPage* tiene las siguientes propiedades:

ctx	Conector con la base de datos.

La clase *RandomEvaluationConfigurationPage* tiene los siguientes métodos:

Save_Click	Guarda la configuración.

Configuración de la Evaluación de Gestos Configurados - (*CustomEvaluationConfigurationPage*)

En la Configuración de la *Evaluación de Gestos Configurados* se puede configurar tanto los perfiles de configuración de detección de gestos como si la evaluación es automática o no.

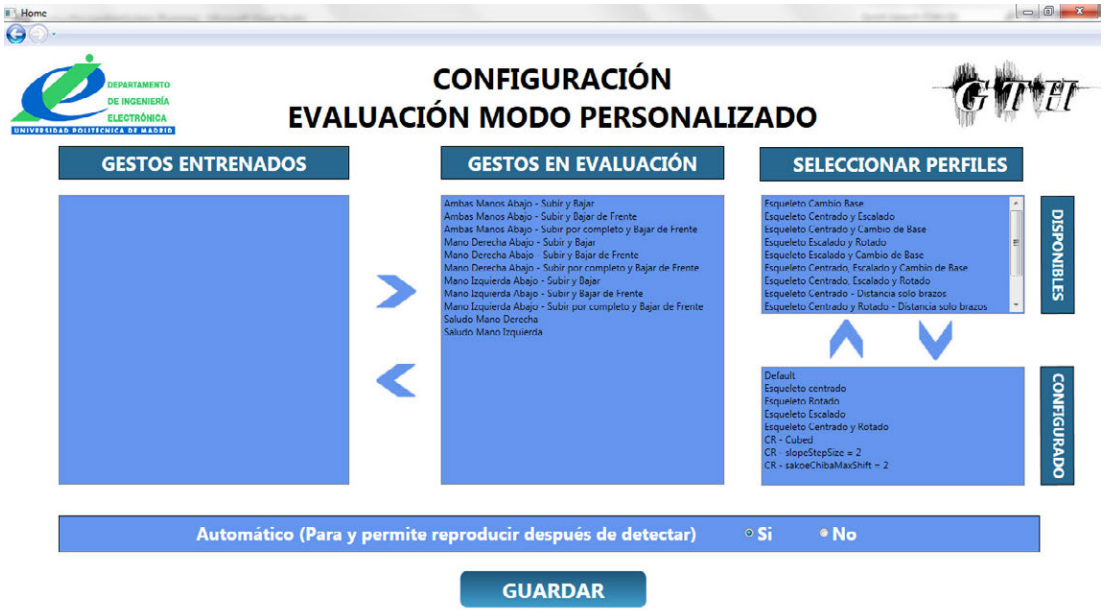


Figura 7.39: Configuración de la Evaluación de Gestos Configurados

Si se habilita la detección automática, en la evaluación se graban gestos continuamente. Si no se habilita la detección automática, el sistema para después de la grabación de cada gesto, lo que permite reproducir o volver a grabar un gesto.

Los perfiles de Configuración para la detección de gestos nos permite seleccionar las configuraciones de detección de gestos que queremos evaluar. Al realizar una evaluación, se calcularán los resultados por cada perfil de configuración. Si hemos configurado una evaluación con 3 perfiles de configuración, se calcularán los resultados de la evaluación aplicando los 3 perfiles. De esta manera podremos comprobar la efectividad de la detección de gestos aplicando un perfil u otro de configuración.

La clase *CustomEvaluationConfigurationPage* contiene las siguientes propiedades:

ctx	Conector con la base de datos.
evaluationGestures	Gestos de la evaluación.
trainingGestures	Gestos entrenados.
availableProfiles	Perfiles de configuración disponibles.
configuredProfiles	Perfiles de configuración de la evaluación.

La clase *CustomEvaluationConfigurationPage* contiene los siguientes métodos:

AddGesture_Click	Añade un gesto de la evaluación.
AddProfile_Click	Añade un perfil de configuración a la evaluación.
DeleteGesture_Click	Quita un gesto de la evaluación.
DeleteProfile_Click	Quita un perfil de configuración a la evaluación.
Save_Click	Guarda la configuración.
UpdateGestures	Actualiza las listas de gestos.
UpdateProfiles	Actualiza las listas de perfiles de configuración.

Configuración de la Detección de Gestos - (*RecognitionConfigurationPage*)

En la Configuración de la *Detección de Gestos*, se puede seleccionar el perfil de configuración para la detección del movimiento y configurar los perfiles de confgurar.

Figura 7.40: Configuración de la Detección de Gestos

El perfil de configuración seleccionado para la detección del movimiento, va a determinar los umbrales de energía y tiempo de la máquina de estados del movimiento y el preprocesamiento del esqueleto.

Los perfiles de configuración se pueden modificar seleccionando el perfil de configuración deseado y haciendo click en *Configurar*. También se pueden añadir o borrar perfiles de configuración. Para añadir un nuevo perfil de configuración, introduciremos el nombre del perfil y haremos click en el botón de *Añadir*. Para borrar un perfil de configuración, seleccionaremos un perfil y haremos click en *Borrar*.

La clase *RecognitionConfigurationPage* contiene las siguientes propiedades:

ctx	Conector con la base de datos.
sysConf	Configuración de la aplicación.

La clase *RecognitionConfigurationPage* contiene los siguientes métodos:

AddProfile_Click	Añade un nuevo perfil de configuración.
DeleteProfile_Click	Borra el perfil de configuración seleccionado.
DetectionGestureListView	Establece el perfil de configuración para la detección del movimiento.
SelectProfile_Click	Configura el perfil de configuración seleccionado.
UpdateLists	Actualiza las lista de perfiles de configuración.

Configuración del Perfil de Configuración - (*ProfileConfigurationPage*)

En la Configuración del *Perfil de Configuración*, se puede acceder a la configuración del **DTW**, a la configuración del **Movimiento** y a la configuración del **Preprocesamiento del Esqueleto**.

Figura 7.41: Configuración del Perfil de Configuración

La clase *ProfileConfigurationPage* contiene las siguientes propiedades:

ConfigurationID	Identificador del perfil de configuración.

La clase *ProfileConfigurationPage* contiene los siguientes métodos:

DTWConfiguration_Click	Accede a la configuración del *DTW*.
MovementConfiguration_Click	Accede a la configuración del *Movimiento*.
PreprocessConfiguration_Click	Accede a la configuración del *Preprocesamiento del esqueleto*.

Configuración DTW - (*DTWConfigurationPage*)

En la configuración DTW, se pueden configurar los parámetros asociados al algoritmo DTW:

- Tipo de distancia empleada al calcular el coste
- Condición inicial de frontera
- Condición final de frontera
- Desplazamiento lateral (paralelogramo de Itakura)
- Desplazamiento en diagonal (paralelogramo de Itakura)
- Pendiente de la Banda de Sakoe-Chiba

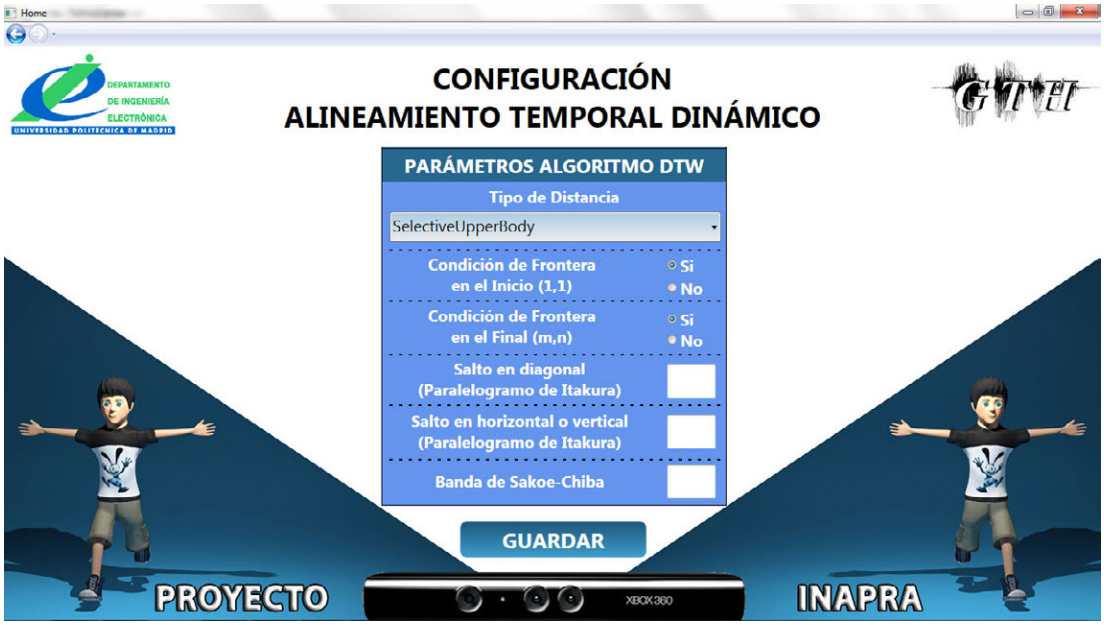


Figura 7.42: Configuración DTW

La clase *DTWConfigurationPage* contiene las siguientes propiedades:

ctx	Conector con la base de datos.
conf	Configuración DTW.

Configuración del Movimiento - (*MovementConfigurationPage*)

En la Configuración del *Movimiento*, se pueden configurar los umbrales de energía y tiempo de la máquina de estados para detectar el movimiento.

Los umbrales de energía y de tiempo son aquellos que marcan la transición de:

- Reposo a Inicio de Movimiento
- Inicio de Movimiento a Reposo
- Inicio de Movimiento a Movimiento
- Movimiento a Inicio de Movimiento
- Movimiento a Fin de Movimiento
- Fin de Movimiento a Movimiento
- Fin de Movimiento a Reposo

Figura 7.43: Configuración del Movimiento

Configuración del Preprocesamiento del Esqueleto - (*PreprocessConfigurationPage*)

En la Configuración del *Preprocesamiento del Esqueleto*, se puede configurar el tipo de preprocesamiento que se aplicará sobre el esqueleto detectado:

- Centrar el esqueleto. Se situará el punto central de la cadera en el punto (0,0,1), siendo el punto (0,0,0) el sensor Kinect.
- Escalar el esqueleto. No afecta al esqueleto al ser coordenadas reales en 3 dimensiones.
- Rotar el esqueleto. Se rotará el esqueleto para que esté mirando siempre de frente. Hay dos tipos de rotación:
    - Rotado sobre el eje Y (vertical que define la columna vertebral)
    - Cambio de Base

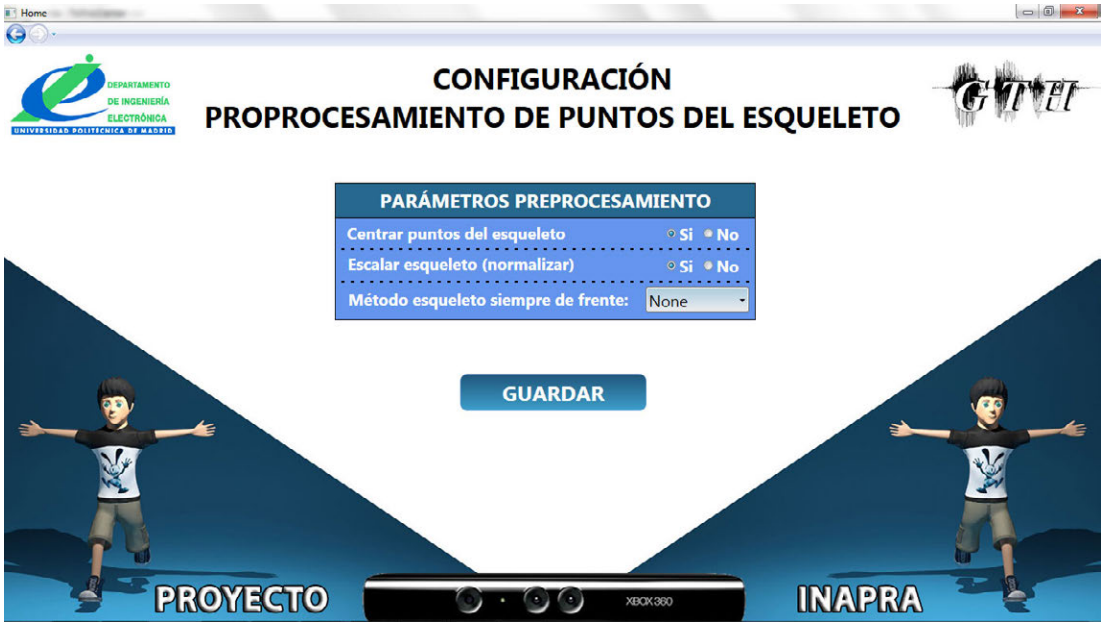


Figura 7.44: Configuración del Preprocesamiento del Esqueleto

La clase *PreprocessConfigurationPage* contiene las siguientes propiedades:

conf	Configuración del preprocesamiento del esqueleto.
ctx	Conector con la base de datos.

7.3. Base de Datos - (*Models*)

El evaluador de gestor se conecta con una base de datos MySQL, donde se guardan los usuarios, configuraciones, entrenamientos y evaluaciones realizadas.

La base de datos se ha creado a través de la tecnología ADO.NET Entity Framework (EF) de Microsoft. Esta simplifica el acceso a los datos permitiendo trabajar directamente con la base de datos en el código. Entity Framework permite interactuar con la base de datos a través de clases y métodos, realizando consultas tanto de lectura como de escritura en la base de datos.

EF proporciona tres modos de describir el modelo de las entidades:

1. Se empieza con una base de datos heredada, y a partir de ella se crea un modelo
2. Se diseñar un modelo en un diseñador
3. Se definen las clases y EF las mapea en la base de datos

Este último modo se conoce como *Code First*, porque lo primero que se hace es escribir el código para describir las entidades, y es el modo que se ha utilizado en este proyecto. Mediante este modo, cada clase se mapea en una tabla de la base de datos, y cada propiedad de la clase, se mapea en una columna de la tabla. Las ventajas de Code First frente a los otros modos es que se puede modificar la estructura de la base de datos, y EF regenera la base de datos automáticamente. Con el primer modo habría que generar una nueva estructura e importarla, y con el segundo modo habría que diseñarla de nuevo con cada cambio realizado. Además, Code First permite que otras personas que tengan acceso al código puedan modificar la estructura de la base de datos simplemente cambiando la clase correspondiente, algo imposible con los otros dos métodos.

Cuando se realiza un cambio en una clase de la base de datos, la estructura cambia y se tiene que volver a regenerar la base de datos. Para ello se usa la clase Initializer, que es donde se define este comportamiento.

```
//public class Initializer :
   DropCreateDatabaseAlways<Context>
public class Initializer :
   DropCreateDatabaseIfModelChanges<Context>
```

Por defecto está habilitado el método *DropCreateDatabaseIfModelChanges* para borrar y regenerar la base de datos sólo si cambia la estructura de la misma. En el caso de que queramos borrar y regenerar la base de datos, no es necesario hacer un cambio en la estructura, sino simplemente descomentar el método *DropCreateDatabaseAlways* y comentar el método *DropCreateDatabaseIfModelChanges*:

```
public class Initializer : DropCreateDatabaseAlways<Context>
//public class Initializer :
   DropCreateDatabaseIfModelChanges<Context>
```

Para acceder a los datos de la base de datos, la clave está en la clase *Context*. En esta clase se definen los *DbSet* o colecciones de objetos que serán accesibles en código. Por defecto añadimos todas las clases del Modelo, ya que nos interesa poder acceder a todos los objetos.

```
public class Context : DbContext
{
        public Context() :
           base("MySqlMembershipConnection") { }

        public DbSet<GestureModel> Gestures { get;
           set; }
  public DbSet<User> Users { get; set; }

        ...
}
```

Como vemos en este extracto de la clase Context, se definen los DbSet para la clase GestureModel y para la clase User. De esta manera, cada vez que queramos consultar todos los gestos de un usuario, simplemente tenemos que hacer:

```
Context ctx = new Context();
List<GestureModel> gestures =
   ctx.Gestures.Where(gesture => gesture.user.name ==
   "Alberto");
```

Si nos fijamos en el constructor, la clase deriva de la clase *DbContext*, y se le pasa parámetro el *connectionString*, *MySqlMembershipConnection*. El *connectionString* se configura en el archivo App.config, y contiene los parámetros de conexión con la base de datos:

```
<?xml version="1.0" encoding="utf-8"?>
<configuration>
<connectionStrings>
    <add name="MySqlMembershipConnection"
       connectionString="Pipe=MySQL; Protocol=pipe;
       Server=127.0.0.1; Database=gestures; User
       Id=grs; Password=gesture;"
       providerName="MySql.Data.MySqlClient" />
  </connectionStrings>
  <system.data>
    <DbProviderFactories>
      <remove invariant="MySql.Data.MySqlClient" />
      <add name="MySQL Data Provider"
         invariant="MySql.Data.MySqlClient"
         description=".Net Framework Data Provider for
         MySQL"
         type="MySql.Data.MySqlClient.MySqlClientFactory,
         MySql.Data, Version=6.7.4.0, Culture=neutral,
         PublicKeyToken=c5687fc88969c44d" />
    </DbProviderFactories>
  </system.data>
</configuration>
```

En este caso la configuración de *connectionStrings* nos indica que la base de datos es *MySQL*, conectada a través de un pipe. En el caso de que quisiéramos acceder a un servidor remoto, habría que conectarlo a través de TCP. En el caso de que el servidor esté en el propio ordenador, el pipe suele ser mejor opción. El servidor es la propia máquina (IP 127.0.0.1), la base de datos se llama *gestures*, el usuario *grs* y la contraseña *gesture*.

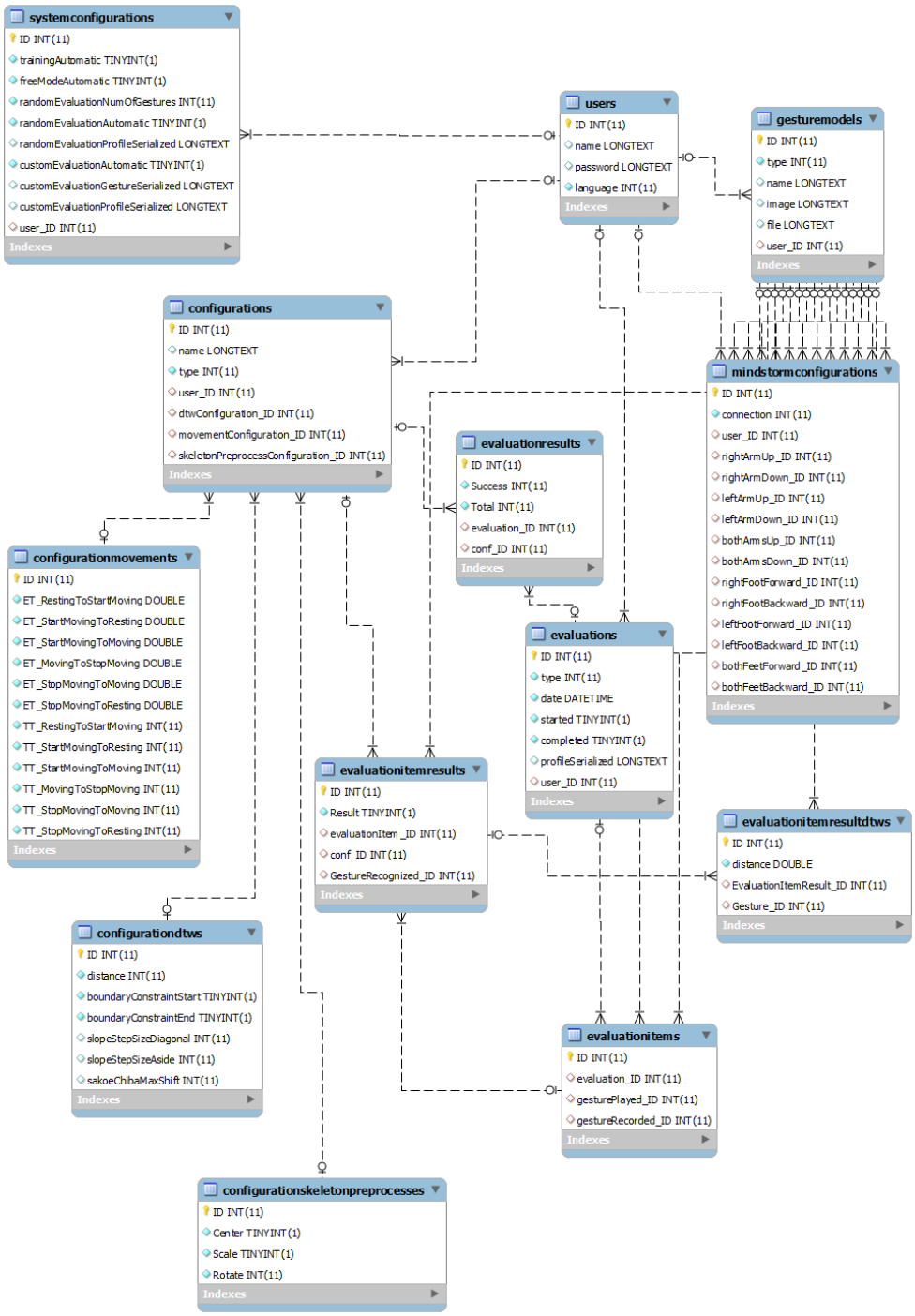


Figura 7.45: Modelo de la base de datos

Initializer

En esta clase se define el comportamiento de la base de datos al arrancar la aplicación. Existen 2 tipos de comportamientos:

- *DropCreateDatabaseAlways* - Cada vez que se inicia la aplicación, se borra y se crea de nuevo la base de datos.
- *DropCreateDatabaseIfModelChanges* - Cada vez que la estructura de la base de datos cambia, se borra y se crea de nuevo la base de datos

El comportamiento configurado es el segundo: sólo se borra la base de datos si ha cambiado la estructura en el código. El primer comportamiento está comentado en el código por si se quiere borrar la base de datos, simplemente es descomentar el método *DropCreateDatabaseAlways* y comentar el método *DropCreateDatabaseIfModelChanges*.

Context

Esta clase define todas las tablas que pueden ser accesibles en el código tanto para lectura como para escritura:

- GestureModel
- User
- Configuration
- ConfigurationDTW
- ConfigurationGestureRecognizer
- ConfigurationMovement
- ConfigurationSkeletonPreprocess
- MindstormConfiguration
- SystemConfiguration
- Evaluation
- EvaluationItem
- EvaluationItemResult
- EvaluationItemResultDTW
- EvaluationResult

7.3.1. Tablas Principales

Gesturemodel

En la tabla Gesturemodel se guardan los gestos entrenados y los gestos realizados para evaluación. Los gestos entrenados son los que posteriormente se usarán en la evaluación como gestos patrón.

En la base de datos no se guardan los esqueletos que componen el gesto, sino la ruta del archivo donde están los esqueletos. En este archivo se guardan todos los frames que componen el gesto.

- **ID** - Identificador único de cada gesto. Clave primaria autoincremental.
- **type** - Gesto de entrenamiento (gestos patrón) o gesto de evaluación.
- **name** - Nombre del gesto. En los gestos de entrenamiento es la acción que realiza el gesto (ambas manos arriba, mano izquierda arriba...). En los gestos realizados en la evaluación, es la fecha y hora a la que se realiza ese gesto.
- **image** - En los gestos de entrenamiento, imagen asociada al gesto. Las imágenes disponibles son imágenes animadas que muestran como realizar el gesto.
- **file** - Archivo donde están guardados los frames correspondientes al gesto.
- **user** - Usuario que ha realizado el gesto.

User

En esta tabla se almacenan los usuarios de la aplicación.

- ID - Identificador único. Clave primaria autoincremental.
- name - Nombre del usuario.
- password - Hash de la contraseña del usuario.
- language - Idioma por defecto del usuario. La aplicación permite seleccionar entre español e inglés.

7.3.2. Tablas de Configuraciones

En la base de datos se han definido las siguientes tablas relativas a Configuraciones:

Configuration

En esta tabla se almacenan los perfiles de configuración de detección de movimiento y detección de gestos.

- **ID** - Identificador único del perfil de configuración. Clave primaria autoincremental.
- **name** - Nombre del perfil de configuración
- **type** - Tipo de perfil: por defecto o personalizado. Cada usuario tiene un perfil por defecto, y puede tener cero o varios perfiles personalizados. Si un tipo de evaluación o entrenamiento requiere un perfil de configuración para la detección del movimiento o para la detección de gestos, pero no hay un perfil configurado, la aplicación cogerá el perfil de configuración por defecto. La configuración por defecto se compone de las configuraciones por defecto del: **Algoritmo DTW** (*dtwConfiguration*), **Reconocedor de Gestos** (*gestureRecognizerConfiguration*), **Detección del Movimiento** (*movementConfiguration*) y **Preprocesado del Esqueleto** (*skeletonPreprocessConfiguration*).
- **user** - Usuario al que pertenece la configuración.
- **gestureRecognizerConfiguration** - Configuración del Reconocedor de Gestos.
- **dtwConfiguration** - Configuración del algoritmo de detección Alineamiento Temporal Dinámico (DTW).
- **movementConfiguration** - Configuración de la máquina de estados para detectar el movimiento.
- **skeletonPreprocessConfiguration** - Configuración del preprocesado del esqueleto para la detección del movimiento y para la detección de gestos.

ConfigurationDTW

En esta tabla se guardan las configuraciones relativas a los parámetros del algoritmo de detección de gestos Alineamiento Temporal Dinámico (DTW).

- **ID** - Identificador único de la configuración DTW.
- **distance** - Tipo de distancia que se utilizará para comparar dos gestos:
  - **Normal** - Todas las componentes del esqueleto contribuyen de la misma manera a la distancia.
  - **Selective** - Sólo algunas componentes del esqueleto contribuyen a la distancia.
  - **Threshold** - Sólo las componentes del esqueleto que superen un umbral de energía contribuyen a la distancia.
  - **SelectiveWithThreshold** - Sólo algunas componentes del esqueleto, y de estas las que superen un umbral de energía, contribuyen a la distancia.
  - **SelectiveUpperBody** - Sólo los hombros, codos, muñecas y manos contribuyen a la distancia.
  - **SelectiveArms** - Sólo los codos, muñecas y manos contribuyen a la distancia.
  - **SelectiveUpperBodySquared** - Sólo los hombros, codos, muñecas y manos contribuyen a la distancia, que se suma elevando al cuadrado cada término.
  - **SelectiveUpperBodyCube** - Sólo los hombros, codos, muñecas y manos contribuyen a la distancia, que se suma elevando al cubo cada término.
- **boundaryConstraintStart** - Se aplica la condición de frontera en (1, 1).
- **boundaryConstraintEnd** - Se aplica la condición de frontera en (m, n).
- **slopeStepSizeDiagonal** - Máxima desplazamiento diagonal (Itakura).

- **slopeStepSizeAside** - Máximo desplazamiento lateral (Itakura).
- **sakoeChibaMaxShift** - Máxima pendiente (Sakoe-Chiba).

ConfigurationGestureRecognizer

En esta tabla se almacenan las configuraciones relativas al reconocedor de gestos.

- **ID** - Identificador único de la configuración del reconocedor de gestos. Clave primaria autoincremental.
- **distanceRecognizerThreshold** - Umbral mínimo entre el gesto más cercano y el segundo más cercano para considerar que el primero es el gesto detectado. Si no se especifica un umbral, se considera que el gesto detectado es el gesto más cercano. Si la distancia entre el gesto más cercano y el segundo más cercano es superior al umbral, el gesto detectado es el primero. Si es menor, se considera que no se ha detectado ningún gesto.

ConfigurationMovement

En esta tabla se almacenan las configuraciones relativas a los parámetros de la máquina estados para detectar el movimiento.

- **ID** - Identificador único de la configuración de movimiento.
- **ET_RestingToStartMoving** - Umbral de energía para pasar del estado *Reposo* a *Posible Inicio de Movimiento*.
- **ET_StartMovingToResting** - Umbral de energía para pasar del estado *Posible Inicio de Movimiento* a *Descanso*.
- **ET_StartMovingToMoving** - Umbral de energía para pasar del estado *Posible Inicio de Movimiento* a *Movimiento*.
- **ET_MovingToStopMoving** - Umbral de energía para pasar del estado *Movimiento* a *Posible Fin de Movimiento*.

- **ET_StopMovingToMoving** - Umbral de energía para pasar del estado *Posible Fin de Movimiento* a *Movimiento*.
- **ET_StopMovingToResting** - Umbral de energía para pasar del estado *Posible Fin de Movimiento* a *Reposo*.
- **TT_RestingToStartMoving** - Umbral de tiempo para pasar del estado *Reposo* a *Posible Inicio de Movimiento*.
- **TT_StartMovingToResting** - Umbral de tiempo para pasar del estado *Posible Inicio de Movimiento* a *Descanso*.
- **TT_StartMovingToMoving** - Umbral de tiempo para pasar del estado *Posible Inicio de Movimiento* a *Movimiento*.
- **TT_MovingToStopMoving** - Umbral de tiempo para pasar del estado *Movimiento* a *Posible Fin de Movimiento*.
- **TT_StopMovingToMoving** - Umbral de tiempo para pasar del estado *Posible Fin de Movimiento* a *Movimiento*.
- **TT_StopMovingToResting** - Umbral de tiempo para pasar del estado *Posible Fin de Movimiento* a *Reposo*.

ConfigurationSkeletonPreprocess

En esta tabla se guardan las configuraciones relativas al preprocesamiento del esqueleto para la detección de movimiento y gestos.

- **ID** - Identificador único de la configuración del preprocesamiento del esqueleto. Clave primaria autoincremental.
- **Center** - Indica si se centra el esqueleto. Al centrar, el centro de la cadera queda en la posición (0,0,1) =>X = 0, Y = 0, Z = 1.
- **Scale** - Indica si se escala el esqueleto.
- **Rotate** - Indica si se rota el esqueleto para que siempre esté mirando de frente. De esta manera los gestos realizados al girar el cuerpo serán detectados como si se estuvieran realizando de frente. El tipo de rotación puede

ser:

- **None** - No se realiza rotación
- **Rotate** - Se rota todo el cuerpo para situar la recta que une los hombros en el eje X (1,0,0).
- **ChangeBase** - Se realiza un cambio de base, siendo la nueva base:
  - **X** - Recta que une los hombros
  - **Y** - Recta que une el centro de la cadera y el centro de los hombros
  - **Z** - Perpendicular a X e Y.

Systemconfiguration

En esta tabla se almacena la configuración de la aplicación para cada usuario.

- **ID** - Identificador único de la configuración del sistema.
- **trainingAutomatic** - Indica si en el entrenamiento, después de cada gesto entrenado, el sistema tiene que parar para que se pueda reproducir o volver a grabar el gesto. Si esta variable es *TRUE*, se entrenarán todos los gestos de manera continua uno tras otro sin parar. Si esta variable es *FALSE*, después de entrenar un gesto, el sistema se parará para que el gesto pueda ser reproducido o se pueda volver a grabar. Para entrenar el siguiente gesto, hay que hacer clic en el botón *SIGUIENTE*. Por defecto la variable *trainingAutomatic* es *TRUE*.
- **freeModeAutomatic** - Indica si en la detección de gestos en modo *FREE MODE*, después de cada gesto reconocido, el sistema tiene que parar para que se pueda reproducir el gesto detectado. Si esta variable es *TRUE*, se reconocerán gestos de manera continua uno tras otro sin parar. Si esta variable es *FALSE*, después de cada gesto detectado, el sistema se parará para que el gesto pueda ser reproducido. Por defecto la variable *freeModeAutomatic* es *TRUE*.

- **randomEvaluationNumOfGestures** - Número de gestos de la evaluación en modo *GESTOS ALEATORIOS*.
- **randomEvaluationAutomatic** - Indica si en la evaluación de *GESTOS ALEATORIOS*, después de cada gesto grabado, el sistema tiene que parar para que se pueda reproducir el gesto grabado. Si esta variable es *TRUE*, se grabarán gestos de manera continua: uno tras otro sin parar. Si esta variable es *FALSE*, después de cada gesto grabado, el sistema se parará para que el gesto pueda ser reproducido. Para grabar un nuevo gesto, hay que hacer clic en el botón *SIGUIENTE*. Por defecto la variable *randomEvaluationAutomatic* es *TRUE*.
- **randomEvaluationProfileSerialized** - Contiene los distintos perfiles de configuración para la evaluación. En una evaluación, se detectan los grabados aplicando cada perfil de configuración. Por tanto, en una evaluación habrá tantas detecciones por gesto como perfiles de configuración.
- **customEvaluationAutomatic** - Indica si en la evaluación de *GESTOS CONFIGURADOS*, después de cada gesto grabado, el sistema tiene que parar para que se pueda reproducir el gesto grabado. Si esta variable es *TRUE*, se grabarán gestos de manera continua: uno tras otro sin parar. Si esta variable es *FALSE*, después de cada gesto grabado, el sistema se parará para que el gesto pueda ser reproducido. Por defecto la variable *customEvaluationAutomatic* es *TRUE*.
- **customEvaluationProfileSerialized** - Contiene los distintos perfiles de configuración para la evaluación. En una evaluación, se detectan los grabados aplicando cada perfil de configuración. Por tanto, en una evaluación habrá tantas detecciones por gesto como perfiles de configuración.
- **user** - Usuario asociado a la configuración del sistema.
- **movementDetectionProfile** - Perfil de configuración usado para detectar el movimiento de un gesto. Los valores de la configuración usados son: umbrales de energía, umbrales de tiempo y preprocesamiento de esqueleto (centrado, escalado y rotado).
- **freeModeDetectionProfile** - Perfil de configuración usado para detectar gestos en el modo *FREE MODE*

7.3.3. Tablas de Evaluaciones

En la base de datos se han definido las siguientes tablas relativas a Evaluaciones:

Evaluations

En esta tabla se almacenan las evaluaciones realizadas.

- **ID** - Identificador único de la evaluación. Clave primaria autoincremental.
- **type** - Indica si se centra el esqueleto. Al centrar, el centro de la cadera queda en la posición (0,0,1) =>X = 0, Y = 0, Z = 1.
- **date** - Fecha exacta en que se realizó la evaluación.
- **started** - Indica si el cálculo de la evaluación ha empezado. Una evaluación consta de dos etapas: en la primera etapa se graban una serie de gestos, y en la segunda etapa, se calculan los resultados de la evaluación. En esta segunda etapa por cada perfil de configuración, se hallan las distancias entre cada gesto grabado y los gestos entrenados.
- **completed** - Indica si el cálculo de la evaluación ha terminado.
- **profileSerialized** - Perfiles de configuración para calcular las distintas distancias en la evaluación.
- **user** - Usuario que realiza la evaluación.

EvaluationItems

En esta tabla se almacenan los elementos de las evaluaciones realizadas. Cada elemento contiene el gesto mostrado y el gestos grabado.

- **ID** - Identificador único del resultado del elemento de la evaluación realizada. Clave primaria autoincremental.
- **evaluation** - Evaluación a la que corresponde el elemento.
- **gesturePlayed** - Gesto mostrado.

- **gestureRecorded** - Gesto grabado.

EvaluationItemResults

En esta tabla se almacenan los resultados de los elementos de las evaluaciones realizadas.

- **ID** - Identificador único del resultado del elemento de la evaluación realizada. Clave primaria autoincremental.
- **Result** - Indica si se ha detectado bien el gesto (1), o si no se ha detectado correctamente el gesto (0).
- **evaluationItem** - Elemento de la evaluación a la que corresponde el resultado.
- **conf** - Perfil de configuración utilizado para obtener los resultados de la evaluación.
- **GestureRecognized** - Gesto reconocido.

EvaluationItemResultsDTW

En esta tabla se almacenan los resultados de aplicar al algoritmo DTW en la detección de gestos en los elementos de las evaluaciones realizadas.

- **ID** - Identificador único del resultado DTW. Clave primaria autoincremental.
- **distance** - Distancia entre el gesto grabado y un gesto entrenado (especificado en el campo Gesture).
- **EvaluationItemResult** - Elemento de la evaluación a la que corresponde el resultado (gesto reproducido y gesto grabado).
- **Gesture** - Gesto entrenado con el que se compara el gesto grabado.

EvaluationResults

En esta tabla se almacenan los resultados de las evaluaciones realizadas.

- **ID** - Identificador único del resultado de la evaluación realizada. Clave primaria autoincremental.
- **Success** - Número de gestos detectados correctamente en la evaluación.
- **Total** - Número total de gestos de la evaluación.
- **evaluation** - Evaluación a la que corresponden los resultados.
- **conf** - Perfil de configuración utilizado para obtener los resultados de la evaluación.

7.3.4. Enumerados

Las tablas de la base de datos admiten un tipo de campo llamado Enumerado. Cada enumerado define un conjunto de valores constantes. De esta manera, podemos definir un enumerado llamado idiomas, y asignar las constantes: español, inglés, francés... Al definir en una tabla el enumerado idiomas, en ese campo podremos insertar cualquiera de los valores asociados a ese enumerado: español, inglés, francés...

Los tipos enumerados definidos en esta base de datos son:

ConfigurationProfileType

Define el tipo de perfil de configuración que puede existir, perfil por defecto (*Default*) o perfil personalizado (*Custom*). Sólo existe un perfil de configuración por defecto, y puede haber cero, uno o varios perfiles personalizados.

En el caso de que en un método se necesite un perfil de configuración y no haya ningún configurado ningún perfil de configuración, se escogerá el perfil por defecto.

ConnectionType

Define el tipo de conexión con el robot Lego Mindstorm: USB o WIFI.

DistanceTypes

Define los distintos tipos de distancia:

- Normal - Todas las componentes del cuerpo aportan de la misma manera a la distancia.
- Selective - Sólo ciertas componentes del cuerpo aportan a la distancia.
- Threshold - Sólo las componentes del cuerpo cuya distancia supere el umbral contribuyen a la distancia.
- SelectiveWithThreshold - Sólo ciertas componentes del cuerpo cuya distancia supere el umbral contribuyen a la distancia.
- *SelectiveUpperBody* - Sólo los hombros, codos, muñecas y manos contribuyen a la distancia.
- *SelectiveArms* - Sólo los codos, muñecas y manos contribuyen a la distancia.
- *SelectiveUpperBodySquared* - Sólo los hombros, codos, muñecas y manos contribuyen a la distancia, siendo esta la suma de las distancias al cuadrado de cada componente.
- *SelectiveUpperBodyCube* - Sólo los hombros, codos, muñecas y manos contribuyen a la distancia, siendo esta la suma de las distancias al cubo de cada componente.

EvaluationType

Define el tipo de evaluación: con gestos aleatorios (*Random*) o con gestos predefinidos (*Custom*).

GestureType

Define el tipo de gesto:

- *TrainingEnabled* - Gesto de entrenamiento habilitado.
- *Evaluation* - Gesto de evaluación.
- *TrainingDisabled* - Gesto de entrenamiento deshabilitado.

Los gestos de entrenamiento habilitados (*TrainingEnabled*) son los gestos que se graban en el entrenamiento y los que sirven posteriormente de modelos para comparar en la detección de gestos.

Los gestos de evaluación (*Evaluation*) son los gestos grabados que se realizan en las evaluaciones.

Los gestos de entrenamiento deshabilitados (*TrainingDisabled*) son los gestos disponibles pero deshabilitados para realizar en el entrenamiento. Si se quiere poder detectar un gesto deshabilitado, en la configuración del entrenamiento se pueden habilitar los gestos *TrainingDisabled* que pasarán al estado *TrainingEnabled*.

Language

Define el idioma de la aplicación: español (*Spanish*) o inglés (*English*). Cada usuario puede configurar el idioma que le resulte más práctico.

RecognitionType

Define el tipo de reconocimiento de órdenes para manejar el robot Lego Mindstorm: reconocimiento por voz (*Speech*), o reconocimiento de gestos (*Visual*).

RotateType

Define el tipo de rotación: sin rotación (*None*), girando sobre el eje Y (*Rotate*) o mediante un cambio de base (*ChangeBase*).

Capítulo 8

Evaluación de Gestos y Análisis de resultados

El evaluador de gestos permite configurar muchos parámetros que influyen en la detección de gestos. Con el objetivo de analizar con detalle esta influencia, se han realizado una serie de evaluaciones con varios perfiles de configuración. En este capítulo analizaremos los resultados de tres tipos de evaluaciones diferentes:

1. Evaluación de 11 gestos con el **cuerpo de frente** a Kinect
2. Evaluación de 11 gestos con el **cuerpo girado 45º** respecto al Kinect
3. Evaluación de 16 gestos: 11 gestos existentes y **5 gestos inexistentes**

8.1. Perfiles de Configuración en cada Evaluación

Cada evaluación ha sido configurada con ocho perfiles de configuración:

1. **Por Defecto**
2. **Esqueleto Centrado**
3. **Esqueleto Centrado y Rotado mediante Cambio de Base**
4. **Esqueleto Centrado y Rotado mediante Giro sobre el eje Y**
5. **Esqueleto Centrado y Rotado - Desplazamiento (Sakoe-Chiba) = 2**
6. **Esqueleto Centrado y Rotado - Desplazamiento (Itakura) = 2**
7. **Esqueleto Centrado y Rotado - Distancia selectiva con umbral**

Todos los perfiles de configuración tienen establecida la distancia como *distancia selectiva - parte superior del cuerpo*. Es decir, que para todos los perfiles de configuración sólo va a influir en la distancia el desplazamiento de **hombros**, **codos**, **muñecas** y **manos**. La explicación de esta configuración es que los gestos entrenados y realizados sólo incluyen movimiento de brazos y manos, por lo que se ha eliminado todo ruido que pudiera interferir en la evaluación.

Las configuraciones **1**, **2**, **3** y **4** sólo cambian el preprocesado que se realiza del esqueleto. Este preprocesado puede ser de tres tipos:

1. **Centrar esqueleto**. El punto central de la cadera se sitúa en (0,0,1)
2. **Escalar esqueleto**. Este preprocesado no realiza ningún cambio en el esqueleto, por lo que no se incluye en la evaluación. Los resultados son los mismos que sin preprocesado.
3. **Rotar esqueleto** para situarlo siempre de frente:
   - **Girando** el esqueleto sobre el eje Y (vertical que define la normal al suelo)
   - **Cambiando la base** del kinect por una basada en la recta entre hombros (eje X), columna vertebral (eje Y) y perpendicular al eje X e Y (eje Z).

Las configuraciones de **5** y **6** son variaciones del algoritmo DTW:

1. Desplazamiento lateral (**Sakoe-Chiba**)
2. Desplazamiento lateral y diagonal (**Paralelogramo de Itakura**)

Y la configuración **7** usa la Distancia Selectiva con Umbral (Pág. 60), que sólo tiene en cuenta las distancias de las componentes del cuerpo que superen un umbral.

$$d_u(a,b) = \sum_i d(p_{ia}, p_{ib}) \quad \forall\ i \in s\ /\ d(p_{ia}, p_{ib}) > u_d$$

$$\mathbf{s} = \text{hombros, codos, muñecas y manos}$$

$$\mathbf{u_d} = \text{umbral de distancia}$$

8.2. Matriz de Confusión

La medida de precisión de cada evaluación se puede realizar comparando el número de aciertos. Sin embargo, este resultado no nos aporta información acerca de los gestos descartados: falsos negativos y verdaderos negativos. Para analizar esta información, hallaremos la matriz de confusión.

Cuadro 8.1: Matriz de Confusión

	Gestos Acertados	**Gestos Fallados**	
Gestos Detectados	Gesto detectado Correctamente (**TP**)	Gesto detectado incorrectamente (**FP**)	**Precisión** $\frac{\sum \text{TP}}{\sum \text{Detectados}}$
Gestos Descartados	Gesto Descartado Incorrectamente (**FN**)	Gesto Descartado Correctamente (**TN**)	**NPV** $\frac{\sum \text{TN}}{\sum \text{Descartados}}$
F_1 $2 \cdot \frac{\text{Cobertura} \cdot \text{Precisión}}{\text{Cobertura} + \text{Precisión}}$	**Cobertura** *Recall* $\frac{\sum \text{TP}}{\sum \text{Cond. Positiva}}$	**Especificidad** *Specificity* $\frac{\sum \text{TN}}{\sum \text{Cond. Negativa}}$	**Acierto** *Accuracy* $\frac{\sum (\text{TP} + \text{TN})}{\sum \text{TP} + \text{TN} + \text{FP} + \text{FN}}$

- Los **Gestos Acertados** se corresponden con la **Condición Positiva**
- Los **Gestos Fallados** se corresponden con la **Condición Negativa**
- Los **Gestos Detectados** se corresponden con el **Resultado Positivo**
- Los **Gestos Descartados** se corresponden con el **Resultado Negativo**
- Los **Gestos Detectados Correctamente** se corresponden con los **Verdaderos Positivos** (**TP**)
- Los **Gestos Detectados Incorrectamente** se corresponden con los **Falsos Positivos** (**FP**)
- Los **Gestos Descartados Incorrectamente** se corresponden con los **Falsos Negativos** (**FN**)

- Los **Gestos Descartados Correctamente** se corresponden con los **Verdaderos Negativos** (**TN**)

8.2.1. Definiciones de la matriz de confusión

- La **precisión** se define como el cociente de la suma de los Gestos Detectados Correctamente (TP) entre los Gestos Detectados (TP + FP):

$$\mathbf{PRE} = \frac{\sum \text{gestos detectados correctamente}}{\sum \text{Gestos Detectados}} = \frac{\sum \text{TP}}{\sum \text{TP + FP}}$$

- La **Negative predictive value** (**NPV**) se define como el cociente de la suma de los Gestos Descartados Correctamente (TN) entre los Gestos Descartados (FN + TN):

$$\mathbf{NPV} = \frac{\sum \text{Gestos Descartados Correctamente}}{\sum \text{Gestos Descartados}} = \frac{\sum \text{TN}}{\sum \text{FN + TN}}$$

- La **cobertura** se define como el cociente de la suma de los gestos detectados correctamente (TP) entre los Gestos Acertados (TP + FP):

$$\mathbf{REC} = \frac{\sum \text{Gestos Detectados Correctamente}}{\sum \text{Gestos Acertados}} = \frac{\sum \text{TP}}{\sum \text{TP + FN}}$$

- La **especificidad** se define como el cociente de la suma de los Gesto Descartado Correctamente (TN) entre los Gestos Fallados (FP + TN):

$$\mathbf{ESP} = \frac{\sum \text{Gesto Descartado Correctamente}}{\sum \text{Gestos Fallados}} = \frac{\sum \text{TP}}{\sum \text{TP + FN}}$$

- El **valor** $-\ \mathbf{F}_1$, o simplemente $\mathbf{F}_1$, es la media armónica de la precisión y la cobertura:

$$\mathbf{F}_1 = 2 \cdot \frac{\text{precisión} \cdot \text{cobertura}}{\text{precisión} + \text{cobertura}} = \frac{2\text{TP}}{(2\text{TP} + \text{FN} + \text{FP})}$$

- El **Acierto** se define como el cociente de la suma de los Gestos Detectados Correctamente (TP) y los Gestos Descartados Correctamente (TN) entre todos los gestos:

$$\mathbf{ACC} = \frac{\text{TP + TN}}{\text{TP + TN + FP + FN}}$$

8.3. Análisis de Evaluaciones

Para cada perfil de configuración, se analizan los resultados de los 3 tipos de evaluaciones realizadas: 11 gestos de frente, 11 gestos con el cuerpo girado 45º y 16 gestos, 5 de ellos inexistentes. Además, se evaluará el impacto de los dos umbrales de rechazo en los valores de la matriz de confusión:

El **Umbral de rechazo** teniendo en cuenta la **distancia del gesto detectado**, $\mathbf{U_{DG}}$, es la distancia máxima que puede tener un gesto detectado para no ser descartado. Si al realizar un gesto, la distancia del gesto detectado es **superior** al umbral de rechazo, se **considera que no hay la suficiente confianza para asegurar que el gesto ha sido detectado correctamente**.

El **Umbral de rechazo** teniendo en cuenta la **distancia entre el gesto detectado y el segundo**, $\mathbf{U_{GDS}}$, es la distancia mínima que puede haber entre el gesto detectado y el segundo más próximo para no ser descartado. Si al realizar un gesto, la distancia del gesto detectado es **inferior** al umbral de rechazo, se **considera que no hay la suficiente confianza para asegurar que el gesto ha sido detectado correctamente**.

Para evitar que la duración del gesto influya en la distancia de los umbrales, las distancias siempre se refieren a **distancias normalizadas por el número de frames**.

En cada evaluación se incluye la siguiente información:

- Tabla que contiene por cada gesto de la evaluación:
  - Distancias de los gestos detectados
  - Distancias entre el gesto detectado y el segundo más próximo
  - Acierto/Fallo
- Gráficas de las distancias de la tabla anterior
- Matrices de confusión para los umbrales de rechazo $\mathbf{U_{DG}}$ y $\mathbf{U_{GDS}}$
- Tabla de valores **F1** y **Acierto** para distintos umbrales de rechazo $\mathbf{U_{DG}}$ y $\mathbf{U_{GDS}}$.

8.3.1. Evaluaciones sin preprocesado

El perfil por defecto no incluye ningún preprocesado. El algoritmo DTW compara cada gesto de la evaluación con todos los gestos entrenados tal cual se grabaron. La distancia de los gestos detectados será más pequeña cuanto menos nos hayamos movido de posición del entrenamiento a la evaluación. Si nos situamos en la misma posición exactamente en el entrenamiento y en la evaluación, la distancia entre el gesto entrenado y el gestos de la evaluación será muy pequeña. Y cuanto más nos alejemos, más aumentará esa distancia.

En el caso de que nos movamos a la izquierda o a la derecha, puede darse el caso de que el movimiento entrenado con un brazo se llegue a confundir con el movimiento realizado en la evaluación con el otro brazo.

Por ejemplo, si subimos el brazo derecho en el entrenamiento, y nos movemos unos 30 o 40 cm a la derecha en la evaluación, ahora el movimiento del brazo izquierdo será prácticamente el mismo que el movimiento del brazo derecho que realizamos en la evaluación.

Figura 8.1: Gestos de levantar brazo derecho y brazo izquierdo en distinta posición

Evaluación de 11 gestos

En la siguiente tabla podemos ver con detalle cuales son los gestos que se han detectado correctamente, las distancias de los gestos detectados, las distancias de cada gesto detectado con el siguiente gesto, y la distancia del gesto detectado erróneamente con el gesto correcto:

GESTO	DISTANCIA GESTO DETECTADO	DISTANCIA GESTO DETECTADO / FRAMES	DISTANCIA ENTRE GESTO DETECTADO Y SEGUNDO	DISTANCIA ENTRE GESTO DETECTADO Y SEGUNDO / FRAMES	ACIERTO
Ambas Manos Abajo - Subir y Bajar	112.84	2.51	1.44	0.03	SI
Ambas Manos Abajo - Subir y Bajar de Frente	114.86	2.50	19.38	0.42	SI
Ambas Manos Abajo - Subir por completo y Bajar de Frente	153.38	2.47	9.79	0.16	NO
Mano Derecha Abajo - Subir y Bajar	116.93	2.49	2.83	0.06	SI
Mano Derecha Abajo - Subir y Bajar de Frente	123.05	2.46	12.75	0.26	SI
Mano Derecha Abajo - Subir por completo y Bajar de Frente	143.50	2.35	6.74	0.11	SI
Mano Derecha Abajo - Subir por completo y Bajar de Frente	97.88	2.39	18.07	0.44	NO
Mano Izquierda Abajo - Subir y Bajar de Frente	96.24	2.41	33.71	0.84	SI
Mano Izquierda Abajo - Subir por completo y Bajar de Frente	133.14	2.38	3.74	0.07	NO
Saludo Mano Derecha	153.03	2.16	30.88	0.43	SI
Saludo Mano Izquierda	165.39	2.30	19.31	0.27	SI

Cuadro 8.2: Evaluación de 11 gestos sin preprocesamiento

Se han detectado correctamente 8 gestos de 11 (porcentaje de acierto del 73 %). Los tres gestos detectados erróneamente son:

- *Ambas Manos Abajo - Subir por completo y Bajar de Frente* se confunde con *Ambas Manos Abajo - Subir y Bajar* con una distancia de casi 10 metros. La razón del fallo es que son gestos muy parecidos, y el gesto detectado erróneamente se ha realizado en una posición más parecida a la del gesto entrenado.

- *Mano Izquierda Abajo - Subir y Bajar*, se confunde con *Mano Izquierda Abajo - Subir y Bajar de Frente*. El problema es este caso es que el gesto correcto se ha realizado en una posición alejada del gesto entrenado, con lo

que la distancia ha aumentado bastante.

- *Mano Izquierda Abajo - Subir por completo y Bajar de Frente*, se confunde con *Mano Izquierda Abajo - Subir y Bajar de Frente*. Las razones del fallo son las mismas que las del primer gesto, siendo la distancia entre el gesto detectado y el gesto correcto muy pequeña, sólo de 3,74 metros.

En general observamos que las distancias de los gestos detectados correctamente son muy grandes. Los valores normales de distancias para gestos idénticos deberían ser menores de 30 metros. En la primera gráfica podemos observamos que más del 50 % de los gestos tiene una distancia superior a 120 metros.

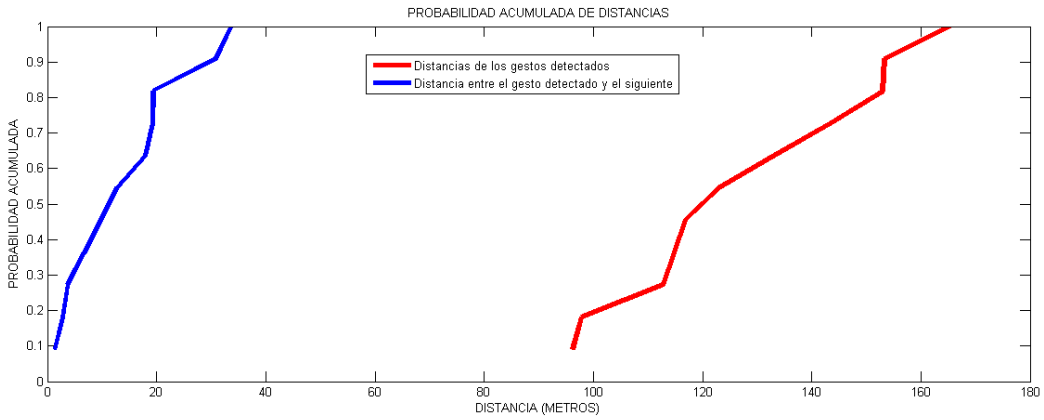


Figura 8.2: Probabilidad acumulada de distancias.

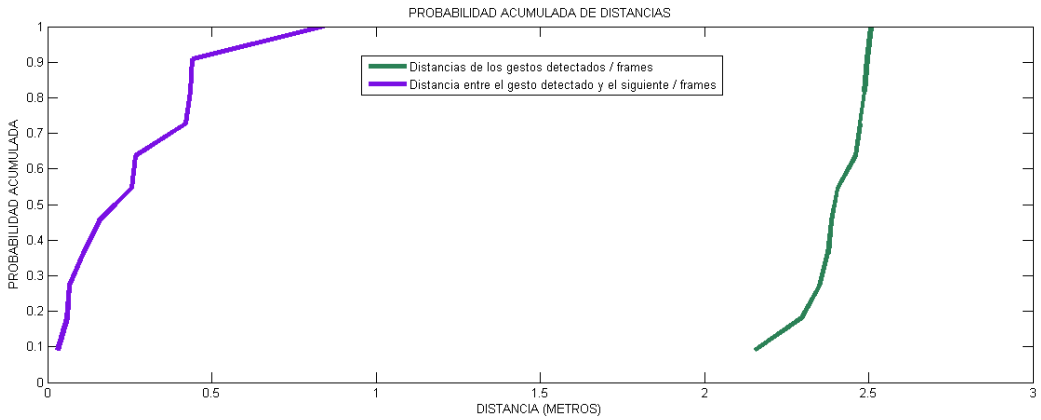


Figura 8.3: Probabilidad acumulada de distancias / frames.

Para más del 50 % de los gestos la distancia entre el gesto detectado y el segundo es menor de 10 metros, frente a una distancia media de los gestos detectados de 120 metros.

Observando la matriz de confusión para los distintos umbrales de rechazo, observamos que los mejores resultados de precisión y cobertura se obtienen sin umbral de rechazo. Para este caso, la precisión es del 73 %, la cobertura del 10 %, el valor $\mathbf{F}_1$ del 84 % y el acierto del 73 %.

Estos resultados nos muestran principalmente dos problemas:

- Los resultados de precisión y cobertura son muy bajos
- Las distancias son muy grandes, con lo que los umbrales descartan todos los gestos

$U_{DG} = 1$

Umbral distancia deteccion de gestos

	GA	GF	
GDT	(TP) 0	(FP) 0	PRE -
GDS	(FN) 8	(TN) 3	NPV 0.27
F_1 -	REC 0	ESP 1	ACC **0.27**

$U_{GDS} = 0.2$

Umbral distancia gesto detectado y segundo

	GA	GF	
GDT	(TP) 5	(FP) 1	PRE 0.83
GDS	(FN) 3	(TN) 2	NPV 0.4
F_1 **0.71**	REC 0.63	ESP 0.67	ACC **0.64**

$U_{DG} = 1$ y $U_{GDS} = 0.2$

	GA	GF	
GDT	(TP) 0	(FP) 1	PRE -
GDS	(FN) 8	(TN) 3	NPV 0.27
F_1 -	REC 0	ESP 1	ACC **0.27**

Sin Umbral

	GA	GF	
GDT	(TP) 8	(FP) 3	PRE 0.73
GDS	(FN) 0	(TN) 0	NPV -
F_1 **0.84**	REC 0.1	ESP 0	ACC **0.73**

Tabla de Valores F-1 y Acierto para distintos Umbrales

U_{DG} [1]	U_{DS} [2]	F_1 [3]	AC [4]
0.02	-	-	0.2727
0.04	-	-	0.2727
0.06	-	-	0.2727
0.08	-	-	0.2727
0.1	-	-	0.2727
0.12	-	-	0.2727
0.14	-	-	0.2727
0.16	-	-	0.2727
0.18	-	-	0.2727
0.2	-	-	0.2727
0.22	-	-	0.2727
0.24	-	-	0.2727
0.26	-	-	0.2727
0.28	-	-	0.2727
0.3	-	-	0.2727
0.32	-	-	0.2727
0.34	-	-	0.2727
0.36	-	-	0.2727
0.38	-	-	0.2727
0.4	-	-	0.2727
0.42	-	-	0.2727
0.44	-	-	0.2727
0.46	-	-	0.2727
0.48	-	-	0.2727
0.5	-	-	0.2727
0.52	-	-	0.2727
0.54	-	-	0.2727
0.56	-	-	0.2727
0.58	-	-	0.2727
0.6	-	-	0.2727
0.62	-	-	0.2727
0.64	-	-	0.2727
0.66	-	-	0.2727
0.68	-	-	0.2727
0.7	-	-	0.2727
0.72	-	-	0.2727
0.74	-	-	0.2727
0.76	-	-	0.2727
0.78	-	-	0.2727
0.8	-	-	0.2727
0.82	-	-	0.2727
0.84	-	-	0.2727
0.86	-	-	0.2727
0.88	-	-	0.2727
0.9	-	-	0.2727
0.92	-	-	0.2727
0.94	-	-	0.2727
0.96	-	-	0.2727
0.98	-	-	0.2727
1	-	-	0.2727

U_{DG} [1]	U_{DS} [2]	F_1 [3]	AC [4]
-	0.02	0.8421	0.7273
-	0.04	0.7778	0.6364
-	0.06	0.7778	0.6364
-	0.08	0.7500	0.6364
-	0.1	0.7500	0.6364
-	0.12	0.6667	0.5455
-	0.14	0.6667	0.5455
-	0.16	0.7143	0.6364
-	0.18	0.7143	0.6364
-	0.2	0.7143	0.6364
-	0.22	0.7143	0.6364
-	0.24	0.7143	0.6364
-	0.26	0.6154	0.5455
-	0.28	0.5000	0.4545
-	0.3	0.5000	0.4545
-	0.32	0.5000	0.4545
-	0.34	0.5000	0.4545
-	0.36	0.5000	0.4545
-	0.38	0.5000	0.4545
-	0.4	0.5000	0.4545
-	0.42	0.5000	0.4545
-	0.44	0.2000	0.2727
-	0.46	0.2222	0.3636
-	0.48	0.2222	0.3636
-	0.5	0.2222	0.3636
-	0.52	0.2222	0.3636
-	0.54	0.2222	0.3636
-	0.56	0.2222	0.3636
-	0.58	0.2222	0.3636
-	0.6	0.2222	0.3636
-	0.62	0.2222	0.3636
-	0.64	0.2222	0.3636
-	0.66	0.2222	0.3636
-	0.68	0.2222	0.3636
-	0.7	0.2222	0.3636
-	0.72	0.2222	0.3636
-	0.74	0.2222	0.3636
-	0.76	0.2222	0.3636
-	0.78	0.2222	0.3636
-	0.8	0.2222	0.3636
-	0.82	0.2222	0.3636
-	0.84	0.2222	0.3636
-	0.86	-	0.2727
-	0.88	-	0.2727
-	0.9	-	0.2727
-	0.92	-	0.2727
-	0.94	-	0.2727
-	0.96	-	0.2727
-	0.98	-	0.2727
-	1	-	0.2727

U_{DG} [1]	U_{DS} [2]	F_1 [3]	AC [4]
0.05	1	-	0.2727
0.05	0.9	-	0.2727
0.05	0.8	-	0.2727
0.1	0.95	-	0.2727
0.1	0.85	-	0.2727
0.1	0.75	-	0.2727
0.15	0.9	-	0.2727
0.15	0.8	-	0.2727
0.15	0.7	-	0.2727
0.2	0.85	-	0.2727
0.2	0.75	-	0.2727
0.2	0.65	-	0.2727
0.25	0.8	-	0.2727
0.25	0.7	-	0.2727
0.25	0.6	-	0.2727
0.3	0.75	-	0.2727
0.3	0.65	-	0.2727
0.3	0.55	-	0.2727
0.35	0.7	-	0.2727
0.35	0.6	-	0.2727
0.35	0.5	-	0.2727
0.4	0.65	-	0.2727
0.4	0.55	-	0.2727
0.4	0.45	-	0.2727
0.45	0.6	-	0.2727
0.45	0.5	-	0.2727
0.45	0.4	-	0.2727
0.5	0.55	-	0.2727
0.5	0.45	-	0.2727
0.5	0.35	-	0.2727
0.55	0.5	-	0.2727
0.55	0.4	-	0.2727
0.55	0.3	-	0.2727
0.6	0.45	-	0.2727
0.6	0.35	-	0.2727
0.6	0.25	-	0.2727
0.65	0.4	-	0.2727
0.65	0.3	-	0.2727
0.65	0.2	-	0.2727
0.7	0.35	-	0.2727
0.7	0.25	-	0.2727
0.7	0.15	-	0.2727
0.75	0.3	-	0.2727
0.75	0.2	-	0.2727
0.75	0.1	-	0.2727
0.8	0.25	-	0.2727
0.8	0.15	-	0.2727
0.8	0.05	-	0.2727
0.85	0.2	-	0.2727
0.85	0.1	-	0.2727

1 - $\mathbf{U_{DG}}$ - Umbral de distancia del gesto detectado
2 - $\mathbf{U_{GDS}}$ - Umbral de distancia entre el gesto detectado y el segundo
3 - $\mathbf{F_1}$ - Valor F-1
4 - **AC** - Acierto

Evaluación de 11 gestos con el cuerpo girado 45º

En esta evaluación observamos que el porcentaje de aciertos es muy bajo, 27 %, y las distancias de los gestos detectados son bastante grandes.

GESTO	DISTANCIA GESTO DETECTADO	DISTANCIA GESTO DETECTADO / FRAMES	DISTANCIA ENTRE GESTO DETECTADO Y SEGUNDO	DISTANCIA ENTRE GESTO DETECTADO Y SEGUNDO / FRAMES	ACIERTO
Ambas Manos Abajo - Subir y Bajar	160.09	2.13	0.28	0.004	NO
Ambas Manos Abajo - Subir y Bajar de Frente	113.90	2.42	10.14	0.216	NO
Ambas Manos Abajo - Subir por completo y Bajar de Frente	113.34	2.31	7.04	0.144	NO
Mano Derecha Abajo - Subir y Bajar	95.19	2.64	14.79	0.411	SI
Mano Derecha Abajo - Subir y Bajar de Frente	78.24	2.06	13.90	0.366	NO
Mano Derecha Abajo - Subir por completo y Bajar de Frente	105.11	2.34	5.76	0.128	NO
Mano Izquierda Abajo - Subir y Bajar	74.52	2.07	9.50	0.264	NO
Mano Izquierda Abajo - Subir y Bajar de Frente	94.07	2.05	12.61	0.274	SI
Mano Izquierda Abajo - Subir por completo y Bajar de Frente	94.09	1.88	12.15	0.243	NO
Saludo Mano Derecha	140.96	1.96	27.65	0.384	SI
Saludo Mano Izquierda	117.94	2.11	3.38	0.060	NO

Cuadro 8.8: Evaluación de 11 gestos con el cuerpo girado 45º sin preprocesamiento

Al girar el cuerpo 45º respecto al entrenamiento, los movimientos laterales se confunden con los movimientos de frente entrenados, y lo mismo a la inversa. Así vemos como *Mano Derecha Abajo - Subir y Bajar de Frente* se confunde con *Mano Derecha Abajo - Subir y Bajar*.

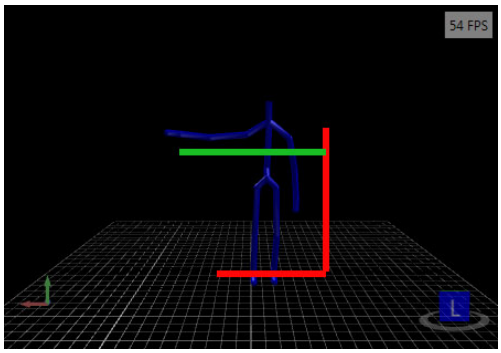


Figura 8.4: Subir y Bajar

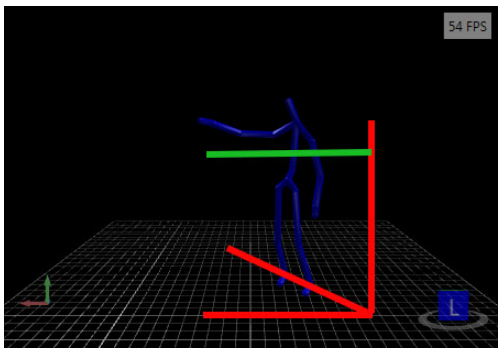


Figura 8.5: Subir y Bajar de Frente

La *Probabilidad acumulada de distancias de gesto detectados* y la *Probabilidad acumulada de distancias de gesto detectado y segundo* nos dan una idea del porcentaje de gestos que están por debajo de una determinada distancia:

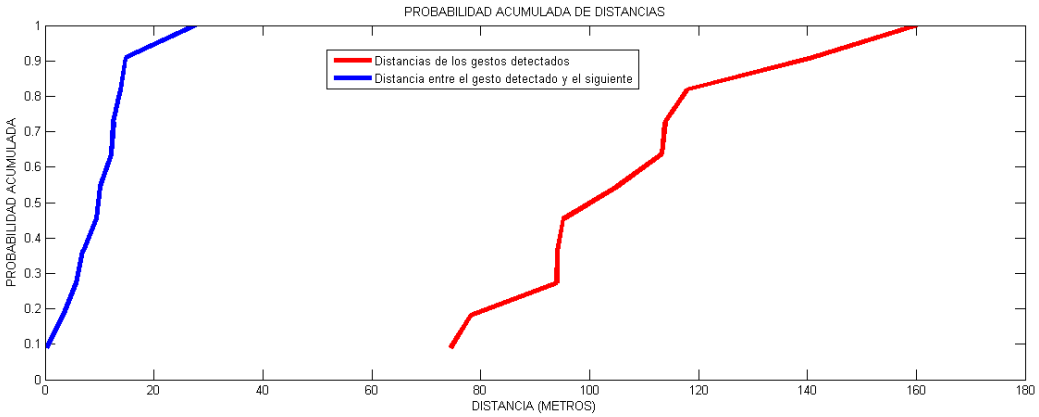


Figura 8.6: Probabilidad acumulada de distancias.

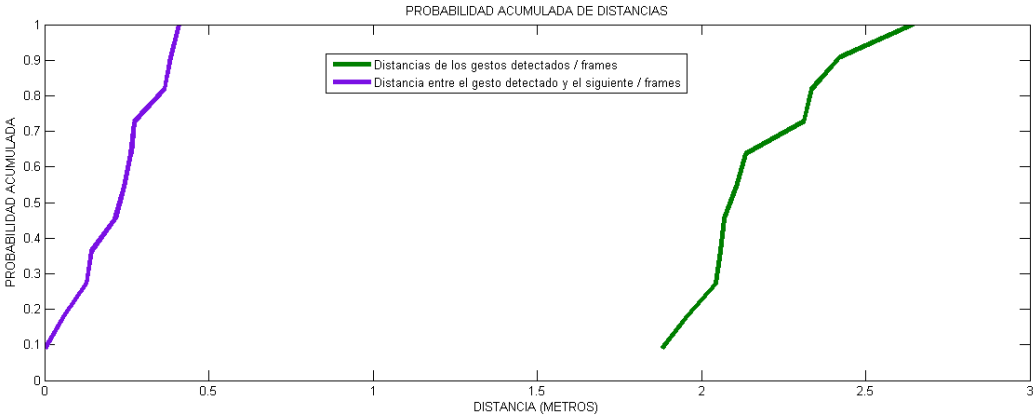


Figura 8.7: Probabilidad acumulada de distancias / frames.

En general observamos que las distancias de los gestos detectados son bastantes grandes. En la primera gráfica podemos observamos que más del 50 % de los gestos tiene una distancia superior a 100 metros.

Las distancias entre cada gesto detectado y el siguiente, el segundo gesto más parecido, tiene valores normales. En la segunda gráfica podemos observar que para más del 50 % de los gestos la distancia entre el gesto detectado y el segundo es menor de 10 metros, frente a una distancia media de los gestos detectados de 100 metros.

Observando las matrices de confusión para los distintos umbrales de rechazo, vemos que el umbral de rechazo $\mathbf{U_{DG}}$ es demasiado bajo y descarta todos los gestos. Sin embargo, el 73 % de estos gestos habían sido detectados incorrectamente, por lo que el umbral de rechazo realmente está funcionando bien. El **acierto** con un umbral $\mathbf{U_{DG}} = \mathbf{1}$ es del 73 %. En cambio, si no utilizamos un umbral de rechazo, el **acierto** es muy inferior, sólo del 27 %.

Con el valor $\mathbf{F_1}$ ocurre lo contrario. Sin umbral de rechazo tiene un valor del 43 %, y con umbral de rechazo no está definido (no hay gestos detectados, todos son descartados).

Sin embargo, en este caso el mejor umbral de rechazo que funciona es $\mathbf{U_{GDS}} = \mathbf{0.2}$. Para este umbral de rechazo el **acierto** es del 64 % y el valor $\mathbf{F_1}$ es del 60 %. Aún así, son valores muy bajos, lo que nos indica que este perfil de configuración no es muy fiable.

$\mathbf{U_{DG}} = \mathbf{1}$

Umbral distancia deteccion de gestos

	GA	GF	
GDT	(TP) 0	(FP) 0	PRE -
GDS	(FN) 3	(TN) 8	NPV 0.73
F_1 **-**	REC 0	ESP 1	ACC **0.73**

$\mathbf{U_{GDS}} = \mathbf{0.2}$

Umbral distancia gesto detectado y segundo

	GA	GF	
GDT	(TP) 3	(FP) 4	PRE 0.43
GDS	(FN) 0	(TN) 4	NPV 1
F_1 **0.6**	REC 1	ESP 0.5	ACC **0.64**

$\mathbf{U_{DG}} = \mathbf{1}$ y $\mathbf{U_{GDS}} = \mathbf{0.2}$

	GA	GF	
GDT	(TP) 0	(FP) 0	PRE -
GDS	(FN) 3	(TN) 8	NPV 0.73
F_1 **-**	REC 0	ESP 1	ACC **0.73**

Sin Umbral

	GA	GF	
GDT	(TP) 3	(FP) 8	PRE 0.27
GDS	(FN) 0	(TN) 0	NPV -
F_1 **0.43**	REC 1	ESP 0	ACC **0.27**

Tabla de Valores F-1 y Acierto para distintos Umbrales

U_{DG} [1]	U_{DS} [2]	F_1 [3]	AC [4]
0.02	-	-	0.7273
0.04	-	-	0.7273
0.06	-	-	0.7273
0.08	-	-	0.7273
0.1	-	-	0.7273
0.12	-	-	0.7273
0.14	-	-	0.7273
0.16	-	-	0.7273
0.18	-	-	0.7273
0.2	-	-	0.7273
0.22	-	-	0.7273
0.24	-	-	0.7273
0.26	-	-	0.7273
0.28	-	-	0.7273
0.3	-	-	0.7273
0.32	-	-	0.7273
0.34	-	-	0.7273
0.36	-	-	0.7273
0.38	-	-	0.7273
0.4	-	-	0.7273
0.42	-	-	0.7273
0.44	-	-	0.7273
0.46	-	-	0.7273
0.48	-	-	0.7273
0.5	-	-	0.7273
0.52	-	-	0.7273
0.54	-	-	0.7273
0.56	-	-	0.7273
0.58	-	-	0.7273
0.6	-	-	0.7273
0.62	-	-	0.7273
0.64	-	-	0.7273
0.66	-	-	0.7273
0.68	-	-	0.7273
0.7	-	-	0.7273
0.72	-	-	0.7273
0.74	-	-	0.7273
0.76	-	-	0.7273
0.78	-	-	0.7273
0.8	-	-	0.7273
0.82	-	-	0.7273
0.84	-	-	0.7273
0.86	-	-	0.7273
0.88	-	-	0.7273
0.9	-	-	0.7273
0.92	-	-	0.7273
0.94	-	-	0.7273
0.96	-	-	0.7273
0.98	-	-	0.7273
1	-	-	0.7273

U_{DG} [1]	U_{DS} [2]	F_1 [3]	AC [4]
-	0.02	0.4615	0.3636
-	0.04	0.4615	0.3636
-	0.06	0.4615	0.3636
-	0.08	0.5000	0.4545
-	0.1	0.5000	0.4545
-	0.12	0.5000	0.4545
-	0.14	0.5455	0.5455
-	0.16	0.6000	0.6364
-	0.18	0.6000	0.6364
-	0.2	0.6000	0.6364
-	0.22	0.6667	0.7273
-	0.24	0.6667	0.7273
-	0.26	0.7500	0.8182
-	0.28	0.6667	0.8182
-	0.3	0.6667	0.8182
-	0.32	0.6667	0.8182
-	0.34	0.6667	0.8182
-	0.36	0.6667	0.8182
-	0.38	0.8000	0.9091
-	0.4	0.5000	0.8182
-	0.42	-	0.7273
-	0.44	-	0.7273
-	0.46	-	0.7273
-	0.48	-	0.7273
-	0.5	-	0.7273
-	0.52	-	0.7273
-	0.54	-	0.7273
-	0.56	-	0.7273
-	0.58	-	0.7273
-	0.6	-	0.7273
-	0.62	-	0.7273
-	0.64	-	0.7273
-	0.66	-	0.7273
-	0.68	-	0.7273
-	0.7	-	0.7273
-	0.72	-	0.7273
-	0.74	-	0.7273
-	0.76	-	0.7273
-	0.78	-	0.7273
-	0.8	-	0.7273
-	0.82	-	0.7273
-	0.84	-	0.7273
-	0.86	-	0.7273
-	0.88	-	0.7273
-	0.9	-	0.7273
-	0.92	-	0.7273
-	0.94	-	0.7273
-	0.96	-	0.7273
-	0.98	-	0.7273
-	1	-	0.7273

U_{DG} [1]	U_{DS} [2]	F_1 [3]	AC [4]
0.05	1	-	0.7273
0.05	0.9	-	0.7273
0.05	0.8	-	0.7273
0.1	0.95	-	0.7273
0.1	0.85	-	0.7273
0.1	0.75	-	0.7273
0.15	0.9	-	0.7273
0.15	0.8	-	0.7273
0.15	0.7	-	0.7273
0.2	0.85	-	0.7273
0.2	0.75	-	0.7273
0.2	0.65	-	0.7273
0.25	0.8	-	0.7273
0.25	0.7	-	0.7273
0.25	0.6	-	0.7273
0.3	0.75	-	0.7273
0.3	0.65	-	0.7273
0.3	0.55	-	0.7273
0.35	0.7	-	0.7273
0.35	0.6	-	0.7273
0.35	0.5	-	0.7273
0.4	0.65	-	0.7273
0.4	0.55	-	0.7273
0.4	0.45	-	0.7273
0.45	0.6	-	0.7273
0.45	0.5	-	0.7273
0.45	0.4	-	0.7273
0.5	0.55	-	0.7273
0.5	0.45	-	0.7273
0.5	0.35	-	0.7273
0.55	0.5	-	0.7273
0.55	0.4	-	0.7273
0.55	0.3	-	0.7273
0.6	0.45	-	0.7273
0.6	0.35	-	0.7273
0.6	0.25	-	0.7273
0.65	0.4	-	0.7273
0.65	0.3	-	0.7273
0.65	0.2	-	0.7273
0.7	0.35	-	0.7273
0.7	0.25	-	0.7273
0.7	0.15	-	0.7273
0.75	0.3	-	0.7273
0.75	0.2	-	0.7273
0.75	0.1	-	0.7273
0.8	0.25	-	0.7273
0.8	0.15	-	0.7273
0.8	0.05	-	0.7273
0.85	0.2	-	0.7273
0.85	0.1	-	0.7273

1 - $\mathbf{U_{DG}}$ - Umbral de distancia del gesto detectado
2 - $\mathbf{U_{GDS}}$ - Umbral de distancia entre el gesto detectado y el segundo
3 - $\mathbf{F_1}$ - Valor F-1
4 - **AC** - Acierto

Evaluación de 16 gestos con 5 gestos inexistentes

En esta evaluación el porcentaje de acierto es del 69 % (11 gestos acertados de 16) debido a que 5 gestos son inexistentes y se van a fallar seguro. Sin embargo, es interesante analizar el umbral de rechazo para ver en qué casos esos gestos van a ser descartados por no superar el umbral de rechazo, y pasar de Falsos Positivos (FP) a Verdaderos Negativos (TN).

GESTO	DISTANCIA GESTO DETECTADO	DISTANCIA GESTO DETECTADO / FRAMES	DISTANCIA ENTRE GESTO DETECTADO Y SEGUNDO	DISTANCIA ENTRE GESTO DETECTADO Y SEGUNDO / FRAMES	ACIERTO
Ambas Manos Abajo - Subir y Bajar	64.39	1.40	27.60	0.60	SI
Ambas Manos Abajo - Subir y Bajar de Frente	32.45	0.72	17.44	0.39	SI
Ambas Manos Abajo - Subir por completo y Bajar de Frente	42.57	0.82	17.03	0.33	SI
Mano Derecha Abajo - Subir y Bajar	26.51	0.66	30.74	0.77	SI
Mano Derecha Abajo - Subir y Bajar de Frente	37.28	0.78	15.64	0.33	SI
Mano Derecha Abajo - Subir por completo y Bajar de Frente	32.19	0.61	14.97	0.28	SI
Mano Izquierda Abajo - Subir y Bajar	23.59	0.58	35.80	0.87	SI
Mano Izquierda Abajo - Subir y Bajar de Frente	23.66	0.58	19.87	0.48	SI
Mano Izquierda Abajo - Subir por completo y Bajar de Frente	36.92	0.75	12.54	0.26	SI
Saludo Mano Derecha	51.30	0.79	33.76	0.52	SI
Saludo Mano Izquierda	53.47	0.89	26.77	0.45	SI
Mano Derecha Abajo - Subir y Bajar Cruzado	55.03	1.22	20.64	0.46	NO
Mano Derecha Abajo - Mover a Izquierda y a Derecha	68.37	1.63	14.59	0.35	NO
Mano Izquierda Abajo - Subir de Frente	53.69	2.24	24.67	1.03	NO
Mano Izquierda Abajo - Mover a Derecha	110.64	2.31	20.03	0.42	NO
Ambas Manos Arriba - Bajar de Frente	50.49	1.80	19.99	0.71	NO

Cuadro 8.14: Evaluación de 16 gestos con perfil sin preprocesamiento

En general observamos que las distancias de los gestos detectados son grandes. En la primera gráfica podemos observamos que más del 50 % de los gestos tiene una distancia superior a 40 metros, y que para más del 50 % de los gestos la distancia entre el gesto detectado y el segundo es menor de 20 metros.

Observando la segunda gráfica nos da una idea del comportamiento de los umbrales en las matrices de confusión. Las distancias entre el gestos detectado y el segundo más próximo divido entre el número de frames de los gestos fallados son: 0,46, 0,35, 1,03, 0,42 y 0,71. Observando la línea morada, más del 50 % de los gestos tienen una distancia superior a 0,46 (8 gestos). Esto nos indica que este umbral de rechazo va a descartar más gestos detectados correctamente que gestos fallados.

Sin embargo, las distancias de los gestos fallados son más grandes que el resto. Observando la línea verde de la segunda gráfica, vemos que si establecemos un umbral alrededor de 1, todos los gestos fallados serán descartados, y sólo estaremos descartando 1 gesto que había sido detectado correctamente.

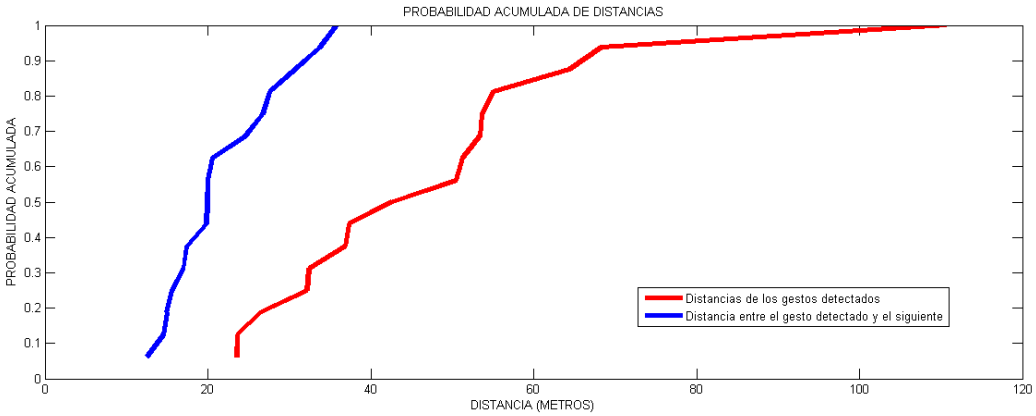


Figura 8.8: Probabilidad acumulada de distancias

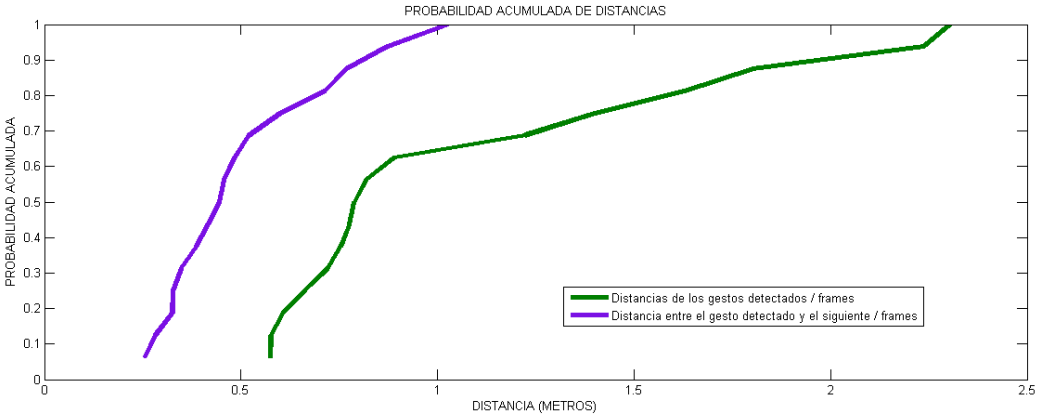


Figura 8.9: Probabilidad acumulada de distancias / frames

Observando las matrices de confusión para los distintos umbrales de rechazo, vemos que el umbral de rechazo $\mathbf{U_{DG}} = 1$ funciona perfectamente.

El **acierto** mejora considerablemente al usar un umbral de rechazo $\mathbf{U_{DG}}$. Sin umbral de rechazo el **acierto** es del 69 %, y con umbral de rechazo $\mathbf{U_{DG}}$ es del 94 %.

Y el mismo comportamiento observamos con el valor $\mathbf{F_1}$. Sin umbral de rechazo el valor $\mathbf{F_1}$ es del 81 %, y con umbral de rechazo $\mathbf{U_{DG}}$ es del 95 %.

$\mathbf{U_{DG} = 1}$

Umbral distancia deteccion de gestos

	GA	GF	
GDT	(TP) 10	(FP) 0	PRE 1
GDS	(FN) 1	(TN) 5	NPV 0.83
F_1 **0.95**	REC 0.91	ESP 1	ACC **0.94**

$\mathbf{U_{GDS} = 0.2}$

Umbral distancia gesto detectado y segundo

	GA	GF	
GDT	(TP) 11	(FP) 5	PRE 0.69
GDS	(FN) 0	(TN) 0	NPV -
F_1 **0.81**	REC 1	ESP 0	ACC **0.69**

$\mathbf{U_{DG} = 1}$ y $\mathbf{U_{GDS} = 0.2}$

	GA	GF	
GDT	(TP) 10	(FP) 0	PRE 1
GDS	(FN) 1	(TN) 5	NPV 0.83
F_1 **0.95**	REC 0.91	ESP 1	ACC **0.94**

Sin Umbral

	GA	GF	
GDT	(TP) 11	(FP) 5	PRE 0.69
GDS	(FN) 0	(TN) 0	NPV -
F_1 **0.81**	REC 1	ESP 0	ACC **0.69**

Tabla de Valores F-1 y Acierto para distintos Umbrales

U_{DG} [1]	U_{DS} [2]	F_1 [3]	AC [4]
0.02	-	-	0.3125
0.04	-	-	0.3125
0.06	-	-	0.3125
0.08	-	-	0.3125
0.1	-	-	0.3125
0.12	-	-	0.3125
0.14	-	-	0.3125
0.16	-	-	0.3125
0.18	-	-	0.3125
0.2	-	-	0.3125
0.22	-	-	0.3125
0.24	-	-	0.3125
0.26	-	-	0.3125
0.28	-	-	0.3125
0.3	-	-	0.3125
0.32	-	-	0.3125
0.34	-	-	0.3125
0.36	-	-	0.3125
0.38	-	-	0.3125
0.4	-	-	0.3125
0.42	-	-	0.3125
0.44	-	-	0.3125
0.46	-	-	0.3125
0.48	-	-	0.3125
0.5	-	-	0.3125
0.52	-	-	0.3125
0.54	-	-	0.3125
0.56	-	-	0.3125
0.58	-	0.3077	0.4375
0.6	-	0.3077	0.4375
0.62	-	0.4286	0.5000
0.64	-	0.4286	0.5000
0.66	-	0.4286	0.5000
0.68	-	0.5333	0.5625
0.7	-	0.5333	0.5625
0.72	-	0.5333	0.5625
0.74	-	0.6250	0.6250
0.76	-	0.7059	0.6875
0.78	-	0.7778	0.7500
0.8	-	0.8421	0.8125
0.82	-	0.9000	0.8750
0.84	-	0.9000	0.8750
0.86	-	0.9000	0.8750
0.88	-	0.9000	0.8750
0.9	-	0.9524	0.9375
0.92	-	0.9524	0.9375
0.94	-	0.9524	0.9375
0.96	-	0.9524	0.9375
0.98	-	0.9524	0.9375
1	-	0.9524	0.9375

U_{DG} [1]	U_{DS} [2]	F_1 [3]	AC [4]
-	0.02	0.8148	0.6875
-	0.04	0.8148	0.6875
-	0.06	0.8148	0.6875
-	0.08	0.8148	0.6875
-	0.1	0.8148	0.6875
-	0.12	0.8148	0.6875
-	0.14	0.8148	0.6875
-	0.16	0.8148	0.6875
-	0.18	0.8148	0.6875
-	0.2	0.8148	0.6875
-	0.22	0.8148	0.6875
-	0.24	0.8148	0.6875
-	0.26	0.7692	0.6250
-	0.28	0.7692	0.6250
-	0.3	0.7200	0.5625
-	0.32	0.7200	0.5625
-	0.34	0.6087	0.4375
-	0.36	0.6364	0.5000
-	0.38	0.6364	0.5000
-	0.4	0.5714	0.4375
-	0.42	0.6000	0.5000
-	0.44	0.6000	0.5000
-	0.46	0.5556	0.5000
-	0.48	0.5556	0.5000
-	0.5	0.4706	0.4375
-	0.52	0.3750	0.3750
-	0.54	0.3750	0.3750
-	0.56	0.3750	0.3750
-	0.58	0.3750	0.3750
-	0.6	0.3750	0.3750
-	0.62	0.2667	0.3125
-	0.64	0.2667	0.3125
-	0.66	0.2667	0.3125
-	0.68	0.2667	0.3125
-	0.7	0.2667	0.3125
-	0.72	0.2857	0.3750
-	0.74	0.2857	0.3750
-	0.76	0.2857	0.3750
-	0.78	0.1538	0.3125
-	0.8	0.1538	0.3125
-	0.82	0.1538	0.3125
-	0.84	0.1538	0.3125
-	0.86	0.1538	0.3125
-	0.88	-	0.2500
-	0.9	-	0.2500
-	0.92	-	0.2500
-	0.94	-	0.2500
-	0.96	-	0.2500
-	0.98	-	0.2500
-	1	-	0.2500

U_{DG} [1]	U_{DS} [2]	F_1 [3]	AC [4]
0.05	1	-	0.3125
0.05	0.9	-	0.3125
0.05	0.8	-	0.3125
0.1	0.95	-	0.3125
0.1	0.85	-	0.3125
0.1	0.75	-	0.3125
0.15	0.9	-	0.3125
0.15	0.8	-	0.3125
0.15	0.7	-	0.3125
0.2	0.85	-	0.3125
0.2	0.75	-	0.3125
0.2	0.65	-	0.3125
0.25	0.8	-	0.3125
0.25	0.7	-	0.3125
0.25	0.6	-	0.3125
0.3	0.75	-	0.3125
0.3	0.65	-	0.3125
0.3	0.55	-	0.3125
0.35	0.7	-	0.3125
0.35	0.6	-	0.3125
0.35	0.5	-	0.3125
0.4	0.65	-	0.3125
0.4	0.55	-	0.3125
0.4	0.45	-	0.3125
0.45	0.6	-	0.3125
0.45	0.5	-	0.3125
0.45	0.4	-	0.3125
0.5	0.55	-	0.3125
0.5	0.45	-	0.3125
0.5	0.35	-	0.3125
0.55	0.5	-	0.3125
0.55	0.4	-	0.3125
0.55	0.3	-	0.3125
0.6	0.45	0.3077	0.4375
0.6	0.35	0.3077	0.4375
0.6	0.25	0.3077	0.4375
0.65	0.4	0.3077	0.4375
0.65	0.3	0.3077	0.4375
0.65	0.2	0.4286	0.5000
0.7	0.35	0.4286	0.5000
0.7	0.25	0.5333	0.5625
0.7	0.15	0.5333	0.5625
0.75	0.3	0.5333	0.5625
0.75	0.2	0.6250	0.6250
0.75	0.1	0.6250	0.6250
0.8	0.25	0.8421	0.8125
0.8	0.15	0.8421	0.8125
0.8	0.05	0.8421	0.8125
0.85	0.2	0.9000	0.8750
0.85	0.1	0.9000	0.8750

1 - $\mathbf{U_{DG}}$ - Umbral de distancia del gesto detectado
2 - $\mathbf{U_{GDS}}$ - Umbral de distancia entre el gesto detectado y el segundo
3 - $\mathbf{F_1}$ - Valor F-1
4 - **AC** - Acierto

8.3.2. Evaluaciones con el esqueleto centrado

El perfil de centrado prepocesa el esqueleto para situar el punto central de cadera en el punto (0,0,1). El punto central de la cadera quedará:

- En la misma posición horizontal que el Kinect (X = 0)
- A la misma altura del kinect (Y = 0)
- A un metro de distancia del Kinect (Z = 0).

Evaluación de 11 gestos

Como vemos en la siguiente tabla, el perfil de centrado obtiene unas distancias de detección de gestos mucho más bajas que con el perfil sin preprocesado. Además, ahora observamos que la distancia de los gestos detectados es inferior a la diferencia de distancia con el segundo gesto más próximo. A mayor fiabilidad, más baja será la distancia de los gestos y mayor será la distancia entre el gesto detectado y el gesto más próximo al detectado.

GESTO	DISTANCIA GESTO DETECTADO	DISTANCIA GESTO DETECTADO / FRAMES	DISTANCIA ENTRE GESTO DETECTADO Y SEGUNDO	DISTANCIA ENTRE GESTO DETECTADO Y SEGUNDO / FRAMES	ACIERTO
Ambas Manos Abajo - Subir y Bajar	17.49	0.39	35.25	0.78	SI
Ambas Manos Abajo - Subir y Bajar de Frente	12.49	0.27	45.62	0.99	SI
Ambas Manos Abajo - Subir por completo y Bajar de Frente	27.12	0.44	33.33	0.54	SI
Mano Derecha Abajo - Subir y Bajar	12.27	0.26	37.85	0.81	SI
Mano Derecha Abajo - Subir y Bajar de Frente	18.81	0.38	28.52	0.57	SI
Mano Derecha Abajo - Subir por completo y Bajar de Frente	21.89	0.36	28.05	0.46	SI
Mano Izquierda Abajo - Subir y Bajar	11.85	0.29	43.27	1.06	SI
Mano Izquierda Abajo - Subir y Bajar de Frente	9.56	0.24	30.24	0.76	SI
Mano Izquierda Abajo - Subir por completo y Bajar de Frente	15.94	0.28	29.03	0.52	SI
Saludo Mano Derecha	28.49	0.40	64.92	0.91	SI
Saludo Mano Izquierda	36.15	0.50	57.25	0.80	SI

Cuadro 8.20: Evaluación de 11 gestos con perfil Centrado

Observando la gráficas de Probabilidad acumulada de distancias de gesto detectados , vemos que más del 50 % de los gestos detectado tiene una distancia inferior a 17 metros. Aunque ahora es un valor que podemos encuadrar dentro de un valor razonable, es mucho menor que la diferencia de distancia con el gesto más próximo, que es superior a 35 metros para más del 50 % de los gesto. Es decir, la distancia entre el gestos detectado y el siguiente es el doble que la distancia de detección.

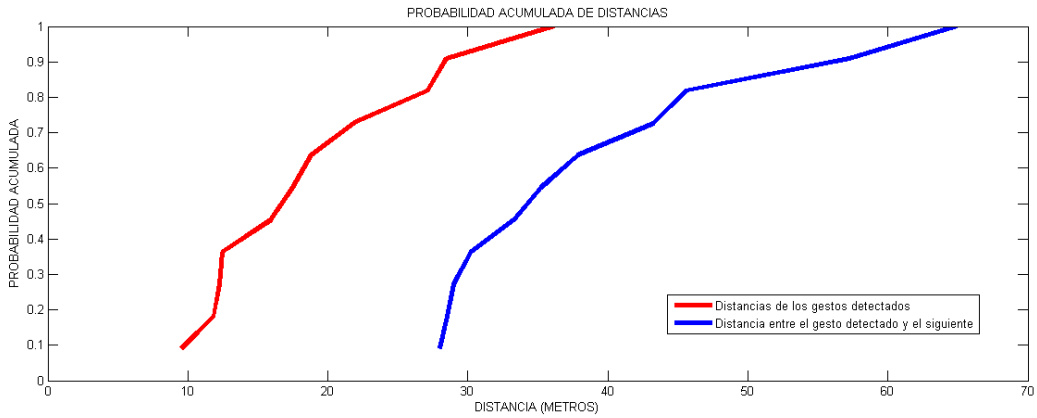


Figura 8.10: Probabilidad acumulada de distancias.

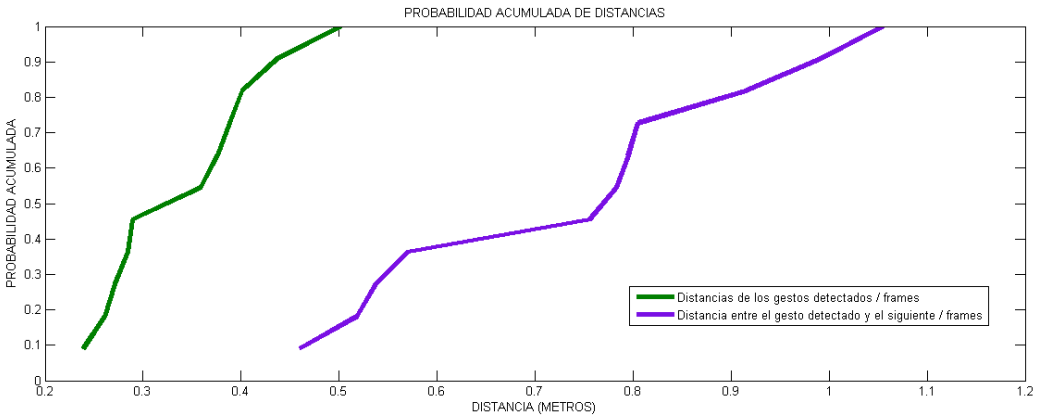


Figura 8.11: Probabilidad acumulada de distancias entre frames.

En esta evaluación todos los gestos han sido detectados correctamente, con unas distancias de detección de gestos bajas. Como vemos en la línea verde de la segunda gráfica, la distancia de los gestos detectados está por debajo de 0,5 ($\mathbf{U_{DG}} = 1 > 0{,}5$. Además, la distancia entre el gesto detectado y el segundo más

próximo está por encima de 0,5 ($\mathbf{U_{GDS}} = 0{,}2 < 0{,}5$. Tanto para $\mathbf{U_{DG}} = 1$ como para $\mathbf{U_{GDS}} = 0{,}2$ no se va a descartar ningún gesto, por lo que los resultados de las matrices de confusión serán los mismos que sin umbral de rechazo.

$\mathbf{U_{DG}} = \mathbf{1}$

Umbral distancia deteccion de gestos

	GA	GF	
GDT	(TP) 11	(FP) 0	PRE 1
GDS	(FN) 0	(TN) 0	NPV -
F_1 **1**	REC 1	ESP -	ACC **1**

$\mathbf{U_{GDS}} = \mathbf{0.2}$

Umbral distancia gesto detectado y segundo

	GA	GF	
GDT	(TP) 11	(FP) 0	PRE 1
GDS	(FN) 0	(TN) 0	NPV -
F_1 **1**	REC 1	ESP -	ACC **1**

$\mathbf{U_{DG}} = \mathbf{1}$ y $\mathbf{U_{GDS}} = \mathbf{0.2}$

	GA	GF	
GDT	(TP) 11	(FP) 0	PRE 1
GDS	(FN) 0	(TN) 0	NPV -
F_1 **1**	REC 1	ESP -	ACC **1**

Sin Umbral

	GA	GF	
GDT	(TP) 11	(FP) 0	PRE 1
GDS	(FN) 0	(TN) 0	NPV -
F_1 **1**	REC 1	ESP -	ACC **1**

Tabla de Valores F-1 y Acierto para distintos Umbrales

U_{DG} [1]	U_{DS} [2]	F_1 [3]	AC [4]
0.02	-	-	0.0000
0.04	-	-	0.0000
0.06	-	-	0.0000
0.08	-	-	0.0000
0.1	-	-	0.0000
0.12	-	-	0.0000
0.14	-	-	0.0000
0.16	-	-	0.0000
0.18	-	-	0.0000
0.2	-	-	0.0000
0.22	-	-	0.0000
0.24	-	0.1667	0.0909
0.26	-	0.1667	0.0909
0.28	-	0.4286	0.2727
0.3	-	0.6250	0.4545
0.32	-	0.6250	0.4545
0.34	-	0.6250	0.4545
0.36	-	0.7059	0.5455
0.38	-	0.7778	0.6364
0.4	-	0.8421	0.7273
0.42	-	0.9000	0.8182
0.44	-	0.9524	0.9091
0.46	-	0.9524	0.9091
0.48	-	0.9524	0.9091
0.5	-	0.9524	0.9091
0.52	-	1.0000	1.0000
0.54	-	1.0000	1.0000
0.56	-	1.0000	1.0000
0.58	-	1.0000	1.0000
0.6	-	1.0000	1.0000
0.62	-	1.0000	1.0000
0.64	-	1.0000	1.0000
0.66	-	1.0000	1.0000
0.68	-	1.0000	1.0000
0.7	-	1.0000	1.0000
0.72	-	1.0000	1.0000
0.74	-	1.0000	1.0000
0.76	-	1.0000	1.0000
0.78	-	1.0000	1.0000
0.8	-	1.0000	1.0000
0.82	-	1.0000	1.0000
0.84	-	1.0000	1.0000
0.86	-	1.0000	1.0000
0.88	-	1.0000	1.0000
0.9	-	1.0000	1.0000
0.92	-	1.0000	1.0000
0.94	-	1.0000	1.0000
0.96	-	1.0000	1.0000
0.98	-	1.0000	1.0000
1	-	1.0000	1.0000

U_{DG} [1]	U_{DS} [2]	F_1 [3]	AC [4]
-	0.02	1.0000	1.0000
-	0.04	1.0000	1.0000
-	0.06	1.0000	1.0000
-	0.08	1.0000	1.0000
-	0.1	1.0000	1.0000
-	0.12	1.0000	1.0000
-	0.14	1.0000	1.0000
-	0.16	1.0000	1.0000
-	0.18	1.0000	1.0000
-	0.2	1.0000	1.0000
-	0.22	1.0000	1.0000
-	0.24	1.0000	1.0000
-	0.26	1.0000	1.0000
-	0.28	1.0000	1.0000
-	0.3	1.0000	1.0000
-	0.32	1.0000	1.0000
-	0.34	1.0000	1.0000
-	0.36	1.0000	1.0000
-	0.38	1.0000	1.0000
-	0.4	1.0000	1.0000
-	0.42	1.0000	1.0000
-	0.44	1.0000	1.0000
-	0.46	0.9524	0.9091
-	0.48	0.9524	0.9091
-	0.5	0.9524	0.9091
-	0.52	0.9000	0.8182
-	0.54	0.8421	0.7273
-	0.56	0.8421	0.7273
-	0.58	0.7778	0.6364
-	0.6	0.7778	0.6364
-	0.62	0.7778	0.6364
-	0.64	0.7778	0.6364
-	0.66	0.7778	0.6364
-	0.68	0.7778	0.6364
-	0.7	0.7778	0.6364
-	0.72	0.7778	0.6364
-	0.74	0.7778	0.6364
-	0.76	0.7059	0.5455
-	0.78	0.7059	0.5455
-	0.8	0.5333	0.3636
-	0.82	0.4286	0.2727
-	0.84	0.4286	0.2727
-	0.86	0.4286	0.2727
-	0.88	0.4286	0.2727
-	0.9	0.4286	0.2727
-	0.92	0.3077	0.1818
-	0.94	0.3077	0.1818
-	0.96	0.3077	0.1818
-	0.98	0.3077	0.1818
-	1	0.1667	0.0909

U_{DG} [1]	U_{DS} [2]	F_1 [3]	AC [4]
0.05	1	-	0.0000
0.05	0.9	-	0.0000
0.05	0.8	-	0.0000
0.1	0.95	-	0.0000
0.1	0.85	-	0.0000
0.1	0.75	-	0.0000
0.15	0.9	-	0.0000
0.15	0.8	-	0.0000
0.15	0.7	-	0.0000
0.2	0.85	-	0.0000
0.2	0.75	-	0.0000
0.2	0.65	-	0.0000
0.25	0.8	-	0.0000
0.25	0.7	0.1667	0.0909
0.25	0.6	0.1667	0.0909
0.3	0.75	0.5333	0.3636
0.3	0.65	0.5333	0.3636
0.3	0.55	0.5333	0.3636
0.35	0.7	0.5333	0.3636
0.35	0.6	0.5333	0.3636
0.35	0.5	0.6250	0.4545
0.4	0.65	0.6250	0.4545
0.4	0.55	0.7059	0.5455
0.4	0.45	0.8421	0.7273
0.45	0.6	0.7059	0.5455
0.45	0.5	0.9000	0.8182
0.45	0.4	0.9524	0.9091
0.5	0.55	0.7778	0.6364
0.5	0.45	0.9524	0.9091
0.5	0.35	0.9524	0.9091
0.55	0.5	0.9524	0.9091
0.55	0.4	1.0000	1.0000
0.55	0.3	1.0000	1.0000
0.6	0.45	1.0000	1.0000
0.6	0.35	1.0000	1.0000
0.6	0.25	1.0000	1.0000
0.65	0.4	1.0000	1.0000
0.65	0.3	1.0000	1.0000
0.65	0.2	1.0000	1.0000
0.7	0.35	1.0000	1.0000
0.7	0.25	1.0000	1.0000
0.7	0.15	1.0000	1.0000
0.75	0.3	1.0000	1.0000
0.75	0.2	1.0000	1.0000
0.75	0.1	1.0000	1.0000
0.8	0.25	1.0000	1.0000
0.8	0.15	1.0000	1.0000
0.8	0.05	1.0000	1.0000
0.85	0.2	1.0000	1.0000
0.85	0.1	1.0000	1.0000

1 - $\mathbf{U_{DG}}$ - Umbral de distancia del gesto detectado
2 - $\mathbf{U_{GDS}}$ - Umbral de distancia entre el gesto detectado y el segundo
3 - $\mathbf{F_1}$ - Valor F-1
4 - **AC** - Acierto

Evaluación de 11 gestos con el cuerpo girado 45º

Esta evaluación, al igual que la misma evaluación realizada sin preprocesamiento, tiene un porcentaje de aciertos muy bajo (36 %) y las distancias de los gestos detectados son muy grandes.

GESTO	DISTANCIA GESTO DETECTADO	DISTANCIA GESTO DETECTADO / FRAMES	DISTANCIA ENTRE GESTO DETECTADO Y SEGUNDO	DISTANCIA ENTRE GESTO DETECTADO Y SEGUNDO / FRAMES	ACIERTO
Ambas Manos Abajo - Subir y Bajar	123.91	1.65	0.16	0.002	NO
Ambas Manos Abajo - Subir y Bajar de Frente	113.95	2.42	0.21	0.004	NO
Ambas Manos Abajo - Subir por completo y Bajar de Frente	111.71	2.28	0.26	0.005	NO
Mano Derecha Abajo - Subir y Bajar	71.94	2.00	24.99	0.694	SI
Mano Derecha Abajo - Subir y Bajar de Frente	48.44	1.27	23.91	0.629	NO
Mano Derecha Abajo - Subir por completo y Bajar de Frente	71.32	1.58	25.83	0.574	NO
Mano Izquierda Abajo - Subir y Bajar	48.81	1.36	8.47	0.235	NO
Mano Izquierda Abajo - Subir y Bajar de Frente	87.38	1.90	9.91	0.215	SI
Mano Izquierda Abajo - Subir por completo y Bajar de Frente	87.27	1.75	8.93	0.179	NO
Saludo Mano Derecha	112.05	1.56	17.87	0.248	SI
Saludo Mano Izquierda	104.25	1.86	0.28	0.005	SI

Cuadro 8.26: Evaluación de 11 gestos con el cuerpo girado 45º con perfil Centrado

La razón de que se detecten erróneamente 7 gestos de 11 es la misma que en la evaluación sin realizar preprocesamiento: al girar el cuerpo 45º respecto al entrenamiento, los movimientos laterales se confunden con los movimientos de frente entrenados, y lo mismo a la inversa.

En las gráficas de la *Probabilidad acumulada de distancias de gesto detectados* y la *Probabilidad acumulada de distancias de gesto detectado y segundo*, observamos que las distancias de los gestos detectados son bastantes grandes. En la primera gráfica podemos observamos que más del 75 % de los gestos tiene una distancia superior a 94 metros.

Las distancias entre el gesto detectado y el segundo gesto más parecido, tiene valores pequeños. Para más del 60 % de los gestos la distancia entre el gesto

detectado y el segundo es menor de 6 metros, frente a una distancia media de los gestos detectados de 98 metros.

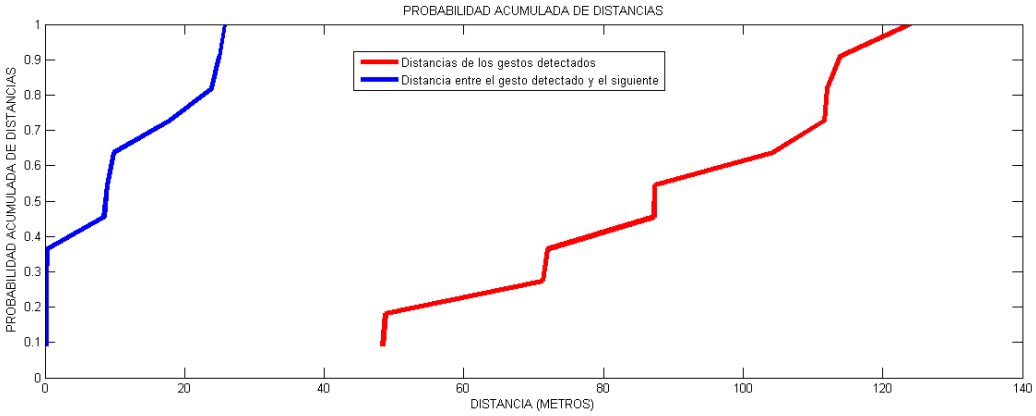


Figura 8.12: Probabilidad acumulada de distancias.

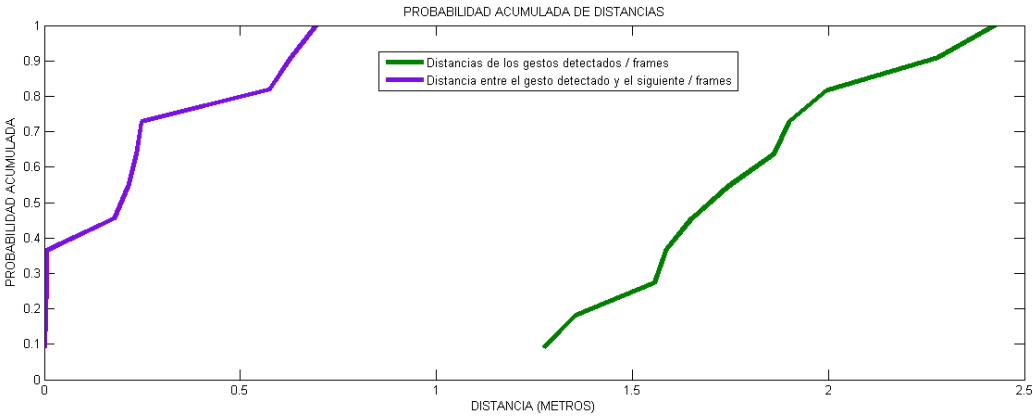


Figura 8.13: Probabilidad acumulada de distancias entre frames.

En las matrices de confusión observamos que, con los umbrales de rechazo, se descartan tanto gestos acertados como gestos fallados. Aunque el **acierto** aumenta al descartar gestos ya que hay más gestos fallados que acertados, los resultados de esta evaluación no son buenos. Los mejores resultados se obtienen con un **acierto** del 64 % y un valor $\mathbf{F}_1$ del 60 % para un umbral $\mathbf{U_{GDS}} = \mathbf{0.2}$.

$U_{DG} = 1$

Umbral distancia deteccion de gestos

	GA	GF	
GDT	(TP) 0	(FP) 0	PRE -
GDS	(FN) 4	(TN) 7	NPV 0.64
F_1 -	REC 0	ESP 1	ACC **0.64**

$U_{GDS} = 0.2$

Umbral distancia gesto detectado y segundo

	GA	GF	
GDT	(TP) 4	(FP) 4	PRE 0.5
GDS	(FN) 1	(TN) 4	NPV 0.8
F_1 **0.6**	REC 0.75	ESP 0.57	ACC **0.64**

$U_{DG} = 1$ y $U_{GDS} = 0.2$

	GA	GF	
GDT	(TP) 0	(FP) 0	PRE -
GDS	(FN) 4	(TN) 7	NPV 0.64
F_1 -	REC 0	ESP 1	ACC **0.64**

Sin Umbral

	GA	GF	
GDT	(TP) 4	(FP) 7	PRE 0.36
GDS	(FN) 0	(TN) 0	NPV -
F_1 **0.53**	REC 1	ESP 0	ACC **0.36**

Tabla de Valores F-1 y Acierto para distintos Umbrales

U_{DG} [1]	U_{DS} [2]	F_1 [3]	AC [4]
0.02	-	-	0.6364
0.04	-	-	0.6364
0.06	-	-	0.6364
0.08	-	-	0.6364
0.1	-	-	0.6364
0.12	-	-	0.6364
0.14	-	-	0.6364
0.16	-	-	0.6364
0.18	-	-	0.6364
0.2	-	-	0.6364
0.22	-	-	0.6364
0.24	-	-	0.6364
0.26	-	-	0.6364
0.28	-	-	0.6364
0.3	-	-	0.6364
0.32	-	-	0.6364
0.34	-	-	0.6364
0.36	-	-	0.6364
0.38	-	-	0.6364
0.4	-	-	0.6364
0.42	-	-	0.6364
0.44	-	-	0.6364
0.46	-	-	0.6364
0.48	-	-	0.6364
0.5	-	-	0.6364
0.52	-	-	0.6364
0.54	-	-	0.6364
0.56	-	-	0.6364
0.58	-	-	0.6364
0.6	-	-	0.6364
0.62	-	-	0.6364
0.64	-	-	0.6364
0.66	-	-	0.6364
0.68	-	-	0.6364
0.7	-	-	0.6364
0.72	-	-	0.6364
0.74	-	-	0.6364
0.76	-	-	0.6364
0.78	-	-	0.6364
0.8	-	-	0.6364
0.82	-	-	0.6364
0.84	-	-	0.6364
0.86	-	-	0.6364
0.88	-	-	0.6364
0.9	-	-	0.6364
0.92	-	-	0.6364
0.94	-	-	0.6364
0.96	-	-	0.6364
0.98	-	-	0.6364
1	-	-	0.6364

U_{DG} [1]	U_{DS} [2]	F_1 [3]	AC [4]
-	0.02	0.5455	0.5455
-	0.04	0.5455	0.5455
-	0.06	0.5455	0.5455
-	0.08	0.5455	0.5455
-	0.1	0.5455	0.5455
-	0.12	0.5455	0.5455
-	0.14	0.5455	0.5455
-	0.16	0.5455	0.5455
-	0.18	0.6000	0.6364
-	0.2	0.6000	0.6364
-	0.22	0.4444	0.5455
-	0.24	0.5000	0.6364
-	0.26	0.2857	0.5455
-	0.28	0.2857	0.5455
-	0.3	0.2857	0.5455
-	0.32	0.2857	0.5455
-	0.34	0.2857	0.5455
-	0.36	0.2857	0.5455
-	0.38	0.2857	0.5455
-	0.4	0.2857	0.5455
-	0.42	0.2857	0.5455
-	0.44	0.2857	0.5455
-	0.46	0.2857	0.5455
-	0.48	0.2857	0.5455
-	0.5	0.2857	0.5455
-	0.52	0.2857	0.5455
-	0.54	0.2857	0.5455
-	0.56	0.2857	0.5455
-	0.58	0.3333	0.6364
-	0.6	0.3333	0.6364
-	0.62	0.3333	0.6364
-	0.64	0.4000	0.7273
-	0.66	0.4000	0.7273
-	0.68	0.4000	0.7273
-	0.7	-	0.6364
-	0.72	-	0.6364
-	0.74	-	0.6364
-	0.76	-	0.6364
-	0.78	-	0.6364
-	0.8	-	0.6364
-	0.82	-	0.6364
-	0.84	-	0.6364
-	0.86	-	0.6364
-	0.88	-	0.6364
-	0.9	-	0.6364
-	0.92	-	0.6364
-	0.94	-	0.6364
-	0.96	-	0.6364
-	0.98	-	0.6364
-	1	-	0.6364

U_{DG} [1]	U_{DS} [2]	F_1 [3]	AC [4]
0.05	1	-	0.6364
0.05	0.9	-	0.6364
0.05	0.8	-	0.6364
0.1	0.95	-	0.6364
0.1	0.85	-	0.6364
0.1	0.75	-	0.6364
0.15	0.9	-	0.6364
0.15	0.8	-	0.6364
0.15	0.7	-	0.6364
0.2	0.85	-	0.6364
0.2	0.75	-	0.6364
0.2	0.65	-	0.6364
0.25	0.8	-	0.6364
0.25	0.7	-	0.6364
0.25	0.6	-	0.6364
0.3	0.75	-	0.6364
0.3	0.65	-	0.6364
0.3	0.55	-	0.6364
0.35	0.7	-	0.6364
0.35	0.6	-	0.6364
0.35	0.5	-	0.6364
0.4	0.65	-	0.6364
0.4	0.55	-	0.6364
0.4	0.45	-	0.6364
0.45	0.6	-	0.6364
0.45	0.5	-	0.6364
0.45	0.4	-	0.6364
0.5	0.55	-	0.6364
0.5	0.45	-	0.6364
0.5	0.35	-	0.6364
0.55	0.5	-	0.6364
0.55	0.4	-	0.6364
0.55	0.3	-	0.6364
0.6	0.45	-	0.6364
0.6	0.35	-	0.6364
0.6	0.25	-	0.6364
0.65	0.4	-	0.6364
0.65	0.3	-	0.6364
0.65	0.2	-	0.6364
0.7	0.35	-	0.6364
0.7	0.25	-	0.6364
0.7	0.15	-	0.6364
0.75	0.3	-	0.6364
0.75	0.2	-	0.6364
0.75	0.1	-	0.6364
0.8	0.25	-	0.6364
0.8	0.15	-	0.6364
0.8	0.05	-	0.6364
0.85	0.2	-	0.6364
0.85	0.1	-	0.6364

1 - $\mathbf{U_{DG}}$ - Umbral de distancia del gesto detectado
2 - $\mathbf{U_{GDS}}$ - Umbral de distancia entre el gesto detectado y el segundo
3 - $\mathbf{F_1}$ - Valor F-1
4 - **AC** - Acierto

Evaluación de 16 gestos con 5 gestos inexistentes

El porcentaje de acierto es del 69 % (11 gestos acertados de 16) debido a que 5 gestos son inexistentes y se van a fallar seguro. Sin embargo, es interesante analizar el umbral de rechazo para ver en qué casos esos gestos van a ser descartados por no superar el umbral de rechazo, y pasar de Falsos Positivos (FP) a Verdaderos Negativos (TN).

GESTO	DISTANCIA GESTO DETECTADO	DISTANCIA GESTO DETECTADO / FRAMES	DISTANCIA ENTRE GESTO DETECTADO Y SEGUNDO	DISTANCIA ENTRE GESTO DETECTADO Y SEGUNDO / FRAMES	ACIERTO
Ambas Manos Abajo - Subir y Bajar	13.40	0.29	39.30	0.85	SI
Ambas Manos Abajo - Subir y Bajar de Frente	11.16	0.25	27.82	0.62	SI
Ambas Manos Abajo - Subir por completo y Bajar de Frente	25.37	0.49	9.17	0.18	SI
Mano Derecha Abajo - Subir y Bajar	9.78	0.24	35.68	0.89	SI
Mano Derecha Abajo - Subir y Bajar de Frente	16.29	0.34	19.89	0.41	SI
Mano Derecha Abajo - Subir por completo y Bajar de Frente	15.39	0.29	9.98	0.19	SI
Mano Izquierda Abajo - Subir y Bajar	9.19	0.22	30.98	0.76	SI
Mano Izquierda Abajo - Subir y Bajar de Frente	6.02	0.15	19.98	0.49	SI
Mano Izquierda Abajo - Subir por completo y Bajar de Frente	11.67	0.24	17.11	0.35	SI
Saludo Mano Derecha	18.72	0.29	41.52	0.64	SI
Saludo Mano Izquierda	24.37	0.41	25.81	0.43	SI
Mano Derecha Abajo - Subir y Bajar Cruzado	38.26	0.85	19.99	0.44	NO
Mano Derecha Abajo - Mover a Izquierda y a Derecha	59.08	1.41	23.87	0.57	NO
Mano Izquierda Abajo - Subir de Frente	39.22	1.63	24.50	1.02	NO
Mano Izquierda Abajo - Mover a Derecha	87.83	1.83	21.22	0.44	NO
Ambas Manos Arriba - Bajar de Frente	37.24	1.33	25.28	0.90	NO

Cuadro 8.32: Evaluación de 16 gestos con perfil Centrado

En la primera gráfica podemos observamos que para un 70 % de los gestos la distancia de los gestos detectados es inferior a la distancia que separa el gesto detectado y el segundo más próximo. Sin embargo, para un 30 % de los gestos, la distancia de los gestos detectados es superior.

Observando la línea morada en la segunda gráfica, vemos que para más del 80 % de los gestos, las distancias entre el gestos detectado y el segundo más próximo divido entre el número de frames son mayores de 0,4. Pero el gran problema es que esos gestos son detectados correctamente. Es decir, los 3 gestos cuyas distancias son menores son gestos detectados correctamente.

Sin embargo, observamos que la distancia de los gestos detectados (línea verde) nos aporta una información muy útil. Todos los gestos fallados tienen una distancia de detección superior a 0,85, mientras que los gestos acertados tienen una distancia de detección inferior a 0,49. En este caso el umbral de rechazo $\mathbf{U_{DG}} = 1$ funcionará perfectamente descartando solamente los gestos fallados. Sin embargo, todavía queda un gesto fallado que no ha sido descartado. El umbral de rechazo $\mathbf{U_{DG}}$ ideal en esta evaluación está entre 0,5 y 0,85.

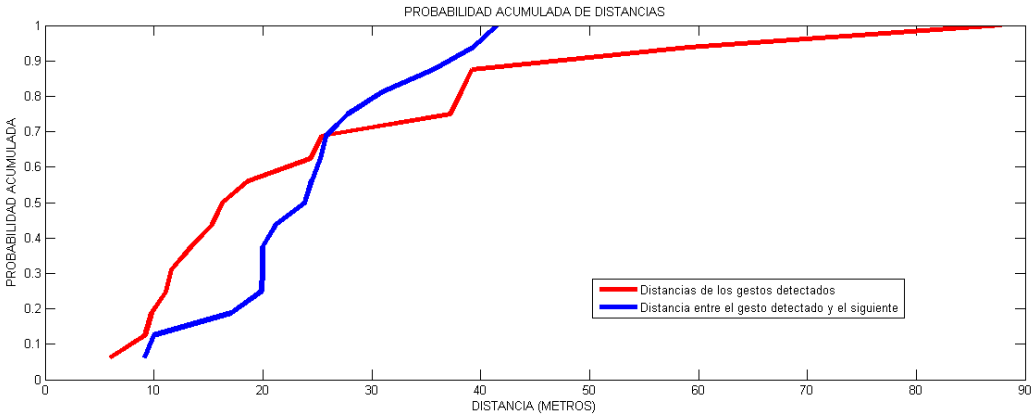


Figura 8.14: Probabilidad acumulada de distancias.

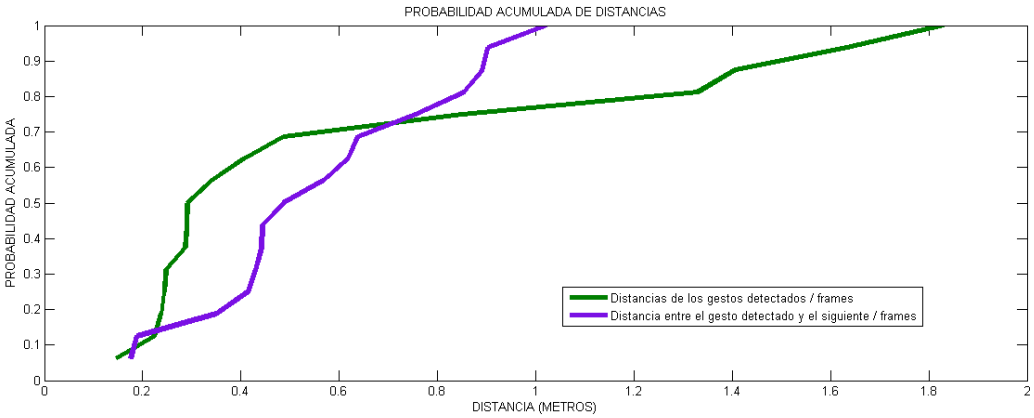


Figura 8.15: Probabilidad acumulada de distancias entre frames.

Observando las matrices de confusión, comprobamos que efectivamente los mejores resultados se obtienen con el umbral de rechazo $\mathbf{U_{DG}} = 1$. El **acierto** es del 94 %, y el valor $\mathbf{F_1}$ del 96 %. Bajando el umbral $\mathbf{U_{DG}}$ por debajo de 0,85 conseguiríamos unos resultados de **acierto** y valor $\mathbf{F_1}$ del 100 %.

$\mathbf{U_{DG}} = 1$

Umbral distancia deteccion de gestos

	GA	GF	
GDT	(TP) 11	(FP) 1	PRE 0.92
GDS	(FN) 0	(TN) 4	NPV 1
F_1 **0.96**	REC 1	ESP 0.8	ACC **0.94**

$\mathbf{U_{GDS}} = 0.2$

Umbral distancia gesto detectado y segundo

	GA	GF	
GDT	(TP) 9	(FP) 5	PRE 0.64
GDS	(FN) 2	(TN) 0	NPV 0
F_1 **0.72**	REC 0.82	ESP 0	ACC **0.56**

$\mathbf{U_{DG}} = 1$ y $\mathbf{U_{GDS}} = 0.2$

	GA	GF	
GDT	(TP) 9	(FP) 1	PRE 0.9
GDS	(FN) 2	(TN) 4	NPV 0.67
F_1 **0.86**	REC 0.82	ESP 0.8	ACC **0.81**

Sin Umbral

	GA	GF	
GDT	(TP) 11	(FP) 5	PRE 0.69
GDS	(FN) 0	(TN) 0	NPV -
F_1 **0.81**	REC 1	ESP 0	ACC **0.69**

Tabla de Valores F-1 y Acierto para distintos Umbrales

U_{DG} [1]	U_{DS} [2]	F_1 [3]	AC [4]
0.02	-	-	0.3125
0.04	-	-	0.3125
0.06	-	-	0.3125
0.08	-	-	0.3125
0.1	-	-	0.3125
0.12	-	-	0.3125
0.14	-	-	0.3125
0.16	-	0.1667	0.3750
0.18	-	0.1667	0.3750
0.2	-	0.1667	0.3750
0.22	-	0.1667	0.3750
0.24	-	0.4286	0.5000
0.26	-	0.6250	0.6250
0.28	-	0.6250	0.6250
0.3	-	0.8421	0.8125
0.32	-	0.8421	0.8125
0.34	-	0.9000	0.8750
0.36	-	0.9000	0.8750
0.38	-	0.9000	0.8750
0.4	-	0.9000	0.8750
0.42	-	0.9524	0.9375
0.44	-	0.9524	0.9375
0.46	-	0.9524	0.9375
0.48	-	0.9524	0.9375
0.5	-	1.0000	1.0000
0.52	-	1.0000	1.0000
0.54	-	1.0000	1.0000
0.56	-	1.0000	1.0000
0.58	-	1.0000	1.0000
0.6	-	1.0000	1.0000
0.62	-	1.0000	1.0000
0.64	-	1.0000	1.0000
0.66	-	1.0000	1.0000
0.68	-	1.0000	1.0000
0.7	-	1.0000	1.0000
0.72	-	1.0000	1.0000
0.74	-	1.0000	1.0000
0.76	-	1.0000	1.0000
0.78	-	1.0000	1.0000
0.8	-	1.0000	1.0000
0.82	-	1.0000	1.0000
0.84	-	1.0000	1.0000
0.86	-	0.9565	0.9375
0.88	-	0.9565	0.9375
0.9	-	0.9565	0.9375
0.92	-	0.9565	0.9375
0.94	-	0.9565	0.9375
0.96	-	0.9565	0.9375
0.98	-	0.9565	0.9375
1	-	0.9565	0.9375

U_{DG} [1]	U_{DS} [2]	F_1 [3]	AC [4]
-	0.02	0.8148	0.6875
-	0.04	0.8148	0.6875
-	0.06	0.8148	0.6875
-	0.08	0.8148	0.6875
-	0.1	0.8148	0.6875
-	0.12	0.8148	0.6875
-	0.14	0.8148	0.6875
-	0.16	0.8148	0.6875
-	0.18	0.7692	0.6250
-	0.2	0.7200	0.5625
-	0.22	0.7200	0.5625
-	0.24	0.7200	0.5625
-	0.26	0.7200	0.5625
-	0.28	0.7200	0.5625
-	0.3	0.7200	0.5625
-	0.32	0.7200	0.5625
-	0.34	0.7200	0.5625
-	0.36	0.6667	0.5000
-	0.38	0.6667	0.5000
-	0.4	0.6667	0.5000
-	0.42	0.6087	0.4375
-	0.44	0.5455	0.3750
-	0.46	0.6000	0.5000
-	0.48	0.6000	0.5000
-	0.5	0.5263	0.4375
-	0.52	0.5263	0.4375
-	0.54	0.5263	0.4375
-	0.56	0.5263	0.4375
-	0.58	0.5556	0.5000
-	0.6	0.5556	0.5000
-	0.62	0.4706	0.4375
-	0.64	0.3750	0.3750
-	0.66	0.3750	0.3750
-	0.68	0.3750	0.3750
-	0.7	0.3750	0.3750
-	0.72	0.3750	0.3750
-	0.74	0.3750	0.3750
-	0.76	0.2667	0.3125
-	0.78	0.2667	0.3125
-	0.8	0.2667	0.3125
-	0.82	0.2667	0.3125
-	0.84	0.2667	0.3125
-	0.86	0.1429	0.2500
-	0.88	0.1429	0.2500
-	0.9	-	0.1875
-	0.92	-	0.2500
-	0.94	-	0.2500
-	0.96	-	0.2500
-	0.98	-	0.2500
-	1	-	0.2500

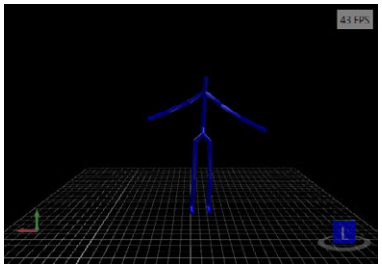
U_{DG} [1]	U_{DS} [2]	F_1 [3]	AC [4]
0.05	1	-	0.3125
0.05	0.9	-	0.3125
0.05	0.8	-	0.3125
0.1	0.95	-	0.3125
0.1	0.85	-	0.3125
0.1	0.75	-	0.3125
0.15	0.9	-	0.3125
0.15	0.8	-	0.3125
0.15	0.7	-	0.3125
0.2	0.85	-	0.3125
0.2	0.75	-	0.3125
0.2	0.65	-	0.3125
0.25	0.8	0.1667	0.3750
0.25	0.7	0.3077	0.4375
0.25	0.6	0.4286	0.5000
0.3	0.75	0.4286	0.5000
0.3	0.65	0.4286	0.5000
0.3	0.55	0.6250	0.6250
0.35	0.7	0.4286	0.5000
0.35	0.6	0.6250	0.6250
0.35	0.5	0.6250	0.6250
0.4	0.65	0.4286	0.5000
0.4	0.55	0.6250	0.6250
0.4	0.45	0.7059	0.6875
0.45	0.6	0.6250	0.6250
0.45	0.5	0.6250	0.6250
0.45	0.4	0.8421	0.8125
0.5	0.55	0.6250	0.6250
0.5	0.45	0.7059	0.6875
0.5	0.35	0.8421	0.8125
0.55	0.5	0.6250	0.6250
0.55	0.4	0.8421	0.8125
0.55	0.3	0.9000	0.8750
0.6	0.45	0.7059	0.6875
0.6	0.35	0.8421	0.8125
0.6	0.25	0.9000	0.8750
0.65	0.4	0.8421	0.8125
0.65	0.3	0.9000	0.8750
0.65	0.2	0.9000	0.8750
0.7	0.35	0.8421	0.8125
0.7	0.25	0.9000	0.8750
0.7	0.15	1.0000	1.0000
0.75	0.3	0.9000	0.8750
0.75	0.2	0.9000	0.8750
0.75	0.1	1.0000	1.0000
0.8	0.25	0.9000	0.8750
0.8	0.15	1.0000	1.0000
0.8	0.05	1.0000	1.0000
0.85	0.2	0.9000	0.8750
0.85	0.1	1.0000	1.0000

1 - $\mathbf{U_{DG}}$ - Umbral de distancia del gesto detectado
2 - $\mathbf{U_{GDS}}$ - Umbral de distancia entre el gesto detectado y el segundo
3 - $\mathbf{F_1}$ - Valor F-1
4 - **AC** - Acierto

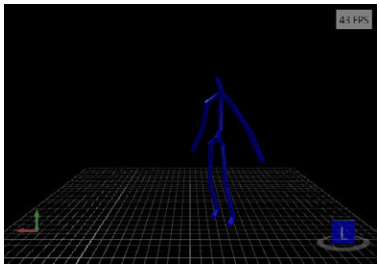
8.3.3. Evaluaciones con el esqueleto centrado y rotado

La operación de rotación tiene como objetivo posicionar el esqueleto siempre de frente al Kinect. De esta manera, al comparar gestos, todos estarán siempre realizados en la misma posición.

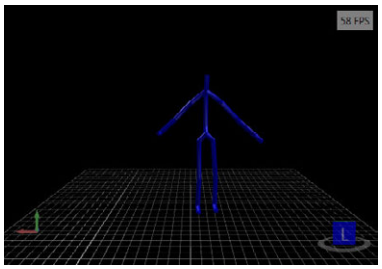
Hay dos tipos de rotación implementados: **Cambio de Base** y **Giro sobre el eje Y**. Las dos rotaciones se analizarán junto a la operación de centrado, ya que tras rotar el esqueleto, este queda de frente pero situado en otra posición. Es necesario realizar una operación de centrado después de rotar para que el esqueleto quede situado en la misma posición.



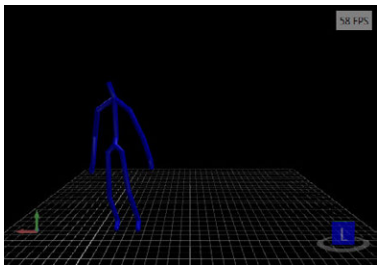
(a) Gesto Entrenado



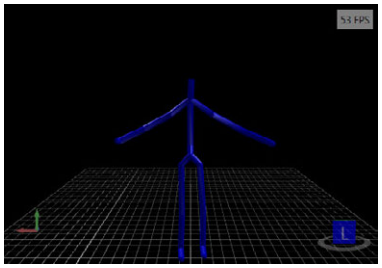
(b) Gesto Evaluación



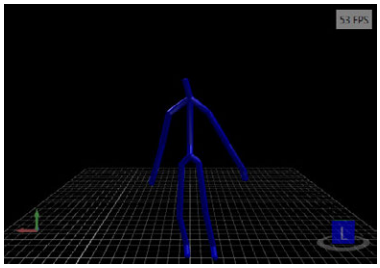
(c) Rotado



(d) Rotado



(e) Centrado y Rotado



(f) Centrado y Rotado

Figura 8.16: Gesto entrenado (izq.) y de la evaluación (derecha) tras centrar y rotar

Evaluaciones con el esqueleto centrado y rotado mediante cambio de base

Este perfil de configuración nos permitirá comparar los gestos como si se hubieran realizado todos de frente y en la misma posición, eliminando las dos principales causas de que se detecten gestos erróneamente con el perfil sin preprocesamiento: gestos realizados en distintas posiciones y gestos realizados con el cuerpo girado.

Evaluación de 11 gestos

Como vemos en la siguiente tabla, el perfil de *centrado y rotado mediante cambio de base* obtiene unos resultados muy parecidos al perfil de sólo centrado.

El porcentaje de aciertos del 100 % y las distancias son muy parecidas: las distancias de los gestos detectados son menores que las distancias entre los gestos detectados y los segundos más próximos.

GESTO	DISTANCIA GESTO DETECTADO	DISTANCIA GESTO DETECTADO / FRAMES	DISTANCIA ENTRE GESTO DETECTADO Y SEGUNDO	DISTANCIA ENTRE GESTO DETECTADO Y SEGUNDO / FRAMES	ACIERTO
Ambas Manos Abajo - Subir y Bajar	11.38	0.25	36.79	0.82	SI
Ambas Manos Abajo - Subir y Bajar de Frente	12.01	0.26	45.12	0.98	SI
Ambas Manos Abajo - Subir por completo y Bajar de Frente	23.26	0.38	41.39	0.67	SI
Mano Derecha Abajo - Subir y Bajar	9.66	0.21	35.97	0.77	SI
Mano Derecha Abajo - Subir y Bajar de Frente	19.35	0.39	20.39	0.41	SI
Mano Derecha Abajo - Subir por completo y Bajar de Frente	23.64	0.39	18.42	0.30	SI
Mano Izquierda Abajo - Subir y Bajar	10.41	0.25	33.95	0.83	SI
Mano Izquierda Abajo - Subir y Bajar de Frente	8.80	0.22	26.99	0.67	SI
Mano Izquierda Abajo - Subir por completo y Bajar de Frente	15.83	0.28	18.66	0.33	SI
Saludo Mano Derecha	27.87	0.39	44.99	0.63	SI
Saludo Mano Izquierda	34.46	0.48	37.96	0.53	SI

Cuadro 8.38: Evaluación de 11 gestos con perfil Centrado y Rotado mediante Cambio de Base

Observando la gráficas de Probabilidad acumulada de distancias de gesto detectados , vemos que más del 50 % de los gestos detectado tiene una distancia inferior a 14 metros.

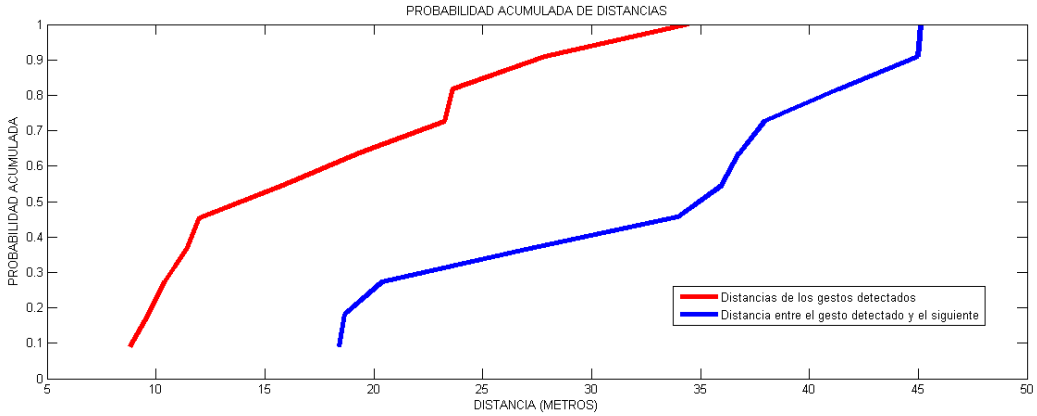


Figura 8.17: Probabilidad acumulada de distancias

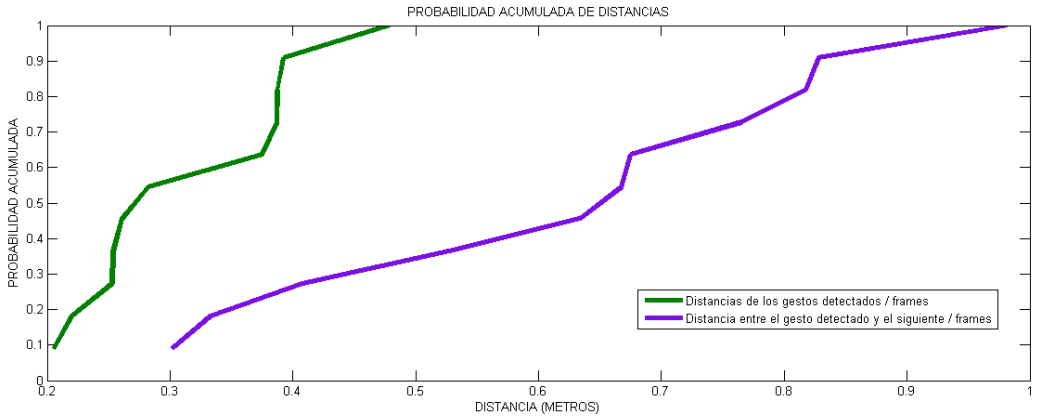


Figura 8.18: Probabilidad acumulada de distancias entre frames

La distancia de los gestos es mucho menor que la diferencia de distancia con el gesto más próximo, que es superior a 30 metros para más del 50 % de los gesto.Es decir, la diferencia de distancia entre el gestos detectado y el siguiente es más del doble que la distancia de detección.

En esta evaluación todos los gestos han sido detectados correctamente, con unas distancias de detección de gestos bajas. Como vemos en la línea verde de la segunda gráfica, la distancia de los gestos detectados está por debajo de 0,6

($\mathbf{U_{DG}} = 1 > 0{,}6$. Además, la distancia entre el gesto detectado y el segundo más próximo está por encima de 0,3 ($\mathbf{U_{GDS}} = 0{,}2 < 0{,}3$. Tanto para $\mathbf{U_{DG}} = 1$ como para $\mathbf{U_{GDS}} = 0{,}2$ no se va a descartar ningún gesto, por lo que los resultados de las matrices de confusión serán los mismos que sin umbral de rechazo.

Para todos los casos los resultados son muy buenos: **acierto** = **100 %** y valor $\mathbf{F_1}$ = **100 %**.

$\mathbf{U_{DG}} = 1$

Umbral distancia deteccion de gestos

	GA	GF	
GDT	(TP) 11	(FP) 0	PRE 1
GDS	(FN) 0	(TN) 0	NPV -
F_1 **1**	REC 1	ESP -	ACC **1**

$\mathbf{U_{GDS}} = 0.2$

Umbral distancia gesto detectado y segundo

	GA	GF	
GDT	(TP) 11	(FP) 0	PRE 1
GDS	(FN) 0	(TN) 0	NPV -
F_1 **1**	REC 1	ESP -	ACC **1**

$\mathbf{U_{DG}} = 1$ y $\mathbf{U_{GDS}} = 0.2$

	GA	GF	
GDT	(TP) 11	(FP) 0	PRE 1
GDS	(FN) 0	(TN) 0	NPV -
F_1 **1**	REC 1	ESP -	ACC **1**

Sin Umbral

	GA	GF	
GDT	(TP) 11	(FP) 0	PRE 1
GDS	(FN) 0	(TN) 0	NPV -
F_1 **1**	REC 1	ESP -	ACC **1**

Tabla de Valores F-1 y Acierto para distintos Umbrales

U_{DG}[1]	U_{DS}[2]	F_1[3]	AC[4]
0.02	-	-	0.0000
0.04	-	-	0.0000
0.06	-	-	0.0000
0.08	-	-	0.0000
0.1	-	-	0.0000
0.12	-	-	0.0000
0.14	-	-	0.0000
0.16	-	-	0.0000
0.18	-	-	0.0000
0.2	-	-	0.0000
0.22	-	0.1667	0.0909
0.24	-	0.3077	0.1818
0.26	-	0.5333	0.3636
0.28	-	0.6250	0.4545
0.3	-	0.7059	0.5455
0.32	-	0.7059	0.5455
0.34	-	0.7059	0.5455
0.36	-	0.7059	0.5455
0.38	-	0.7778	0.6364
0.4	-	0.9524	0.9091
0.42	-	0.9524	0.9091
0.44	-	0.9524	0.9091
0.46	-	0.9524	0.9091
0.48	-	1.0000	1.0000
0.5	-	1.0000	1.0000
0.52	-	1.0000	1.0000
0.54	-	1.0000	1.0000
0.56	-	1.0000	1.0000
0.58	-	1.0000	1.0000
0.6	-	1.0000	1.0000
0.62	-	1.0000	1.0000
0.64	-	1.0000	1.0000
0.66	-	1.0000	1.0000
0.68	-	1.0000	1.0000
0.7	-	1.0000	1.0000
0.72	-	1.0000	1.0000
0.74	-	1.0000	1.0000
0.76	-	1.0000	1.0000
0.78	-	1.0000	1.0000
0.8	-	1.0000	1.0000
0.82	-	1.0000	1.0000
0.84	-	1.0000	1.0000
0.86	-	1.0000	1.0000
0.88	-	1.0000	1.0000
0.9	-	1.0000	1.0000
0.92	-	1.0000	1.0000
0.94	-	1.0000	1.0000
0.96	-	1.0000	1.0000
0.98	-	1.0000	1.0000
1	-	1.0000	1.0000

U_{DG}[1]	U_{DS}[2]	F_1[3]	AC[4]
-	0.02	1.0000	1.0000
-	0.04	1.0000	1.0000
-	0.06	1.0000	1.0000
-	0.08	1.0000	1.0000
-	0.1	1.0000	1.0000
-	0.12	1.0000	1.0000
-	0.14	1.0000	1.0000
-	0.16	1.0000	1.0000
-	0.18	1.0000	1.0000
-	0.2	1.0000	1.0000
-	0.22	1.0000	1.0000
-	0.24	1.0000	1.0000
-	0.26	1.0000	1.0000
-	0.28	1.0000	1.0000
-	0.3	1.0000	1.0000
-	0.32	0.9524	0.9091
-	0.34	0.9000	0.8182
-	0.36	0.9000	0.8182
-	0.38	0.9000	0.8182
-	0.4	0.9000	0.8182
-	0.42	0.8421	0.7273
-	0.44	0.8421	0.7273
-	0.46	0.8421	0.7273
-	0.48	0.8421	0.7273
-	0.5	0.8421	0.7273
-	0.52	0.8421	0.7273
-	0.54	0.7778	0.6364
-	0.56	0.7778	0.6364
-	0.58	0.7778	0.6364
-	0.6	0.7778	0.6364
-	0.62	0.7778	0.6364
-	0.64	0.7059	0.5455
-	0.66	0.7059	0.5455
-	0.68	0.5333	0.3636
-	0.7	0.5333	0.3636
-	0.72	0.5333	0.3636
-	0.74	0.5333	0.3636
-	0.76	0.5333	0.3636
-	0.78	0.4286	0.2727
-	0.8	0.4286	0.2727
-	0.82	0.3077	0.1818
-	0.84	0.1667	0.0909
-	0.86	0.1667	0.0909
-	0.88	0.1667	0.0909
-	0.9	0.1667	0.0909
-	0.92	0.1667	0.0909
-	0.94	0.1667	0.0909
-	0.96	0.1667	0.0909
-	0.98	0.1667	0.0909
-	1	-	0.0000

U_{DG}[1]	U_{DS}[2]	F_1[3]	AC[4]
0.05	1	-	0.0000
0.05	0.9	-	0.0000
0.05	0.8	-	0.0000
0.1	0.95	-	0.0000
0.1	0.85	-	0.0000
0.1	0.75	-	0.0000
0.15	0.9	-	0.0000
0.15	0.8	-	0.0000
0.15	0.7	-	0.0000
0.2	0.85	-	0.0000
0.2	0.75	-	0.0000
0.2	0.65	-	0.0000
0.25	0.8	-	0.0000
0.25	0.7	0.1667	0.0909
0.25	0.6	0.3077	0.1818
0.3	0.75	0.5333	0.3636
0.3	0.65	0.6250	0.4545
0.3	0.55	0.6250	0.4545
0.35	0.7	0.5333	0.3636
0.35	0.6	0.6250	0.4545
0.35	0.5	0.6250	0.4545
0.4	0.65	0.7059	0.5455
0.4	0.55	0.7778	0.6364
0.4	0.45	0.7778	0.6364
0.45	0.6	0.7778	0.6364
0.45	0.5	0.7778	0.6364
0.45	0.4	0.8421	0.7273
0.5	0.55	0.7778	0.6364
0.5	0.45	0.8421	0.7273
0.5	0.35	0.9000	0.8182
0.55	0.5	0.8421	0.7273
0.55	0.4	0.9000	0.8182
0.55	0.3	1.0000	1.0000
0.6	0.45	0.8421	0.7273
0.6	0.35	0.9000	0.8182
0.6	0.25	1.0000	1.0000
0.65	0.4	0.9000	0.8182
0.65	0.3	1.0000	1.0000
0.65	0.2	1.0000	1.0000
0.7	0.35	0.9000	0.8182
0.7	0.25	1.0000	1.0000
0.7	0.15	1.0000	1.0000
0.75	0.3	1.0000	1.0000
0.75	0.2	1.0000	1.0000
0.75	0.1	1.0000	1.0000
0.8	0.25	1.0000	1.0000
0.8	0.15	1.0000	1.0000
0.8	0.05	1.0000	1.0000
0.85	0.2	1.0000	1.0000
0.85	0.1	1.0000	1.0000

1 - $\mathbf{U_{DG}}$ - Umbral de distancia del gesto detectado
2 - $\mathbf{U_{GDS}}$ - Umbral de distancia entre el gesto detectado y el segundo
3 - $\mathbf{F_1}$ - Valor F-1
4 - **AC** - Acierto

Evaluación de 11 gestos con el cuerpo girado 45º

Esta evaluación, al contrario que la misma evaluación realizada con sólo centrado, tiene un porcentaje de aciertos muy alto, del 91 %. Además, las distancias de detección de gestos ya son valores razonables.

GESTO	DISTANCIA GESTO DETECTADO	DISTANCIA GESTO DETECTADO / FRAMES	DISTANCIA ENTRE GESTO DETECTADO Y SEGUNDO	DISTANCIA ENTRE GESTO DETECTADO Y SEGUNDO / FRAMES	ACIERTO
Ambas Manos Abajo - Subir y Bajar	47.12	0.63	25.88	0.35	SI
Ambas Manos Abajo - Subir y Bajar de Frente	45.17	0.96	2.51	0.05	SI
Ambas Manos Abajo - Subir por completo y Bajar de Frente	37.76	0.77	7.15	0.15	SI
Mano Derecha Abajo - Subir y Bajar	13.29	0.37	33.71	0.94	SI
Mano Derecha Abajo - Subir y Bajar de Frente	24.01	0.63	3.28	0.09	SI
Mano Derecha Abajo - Subir por completo y Bajar de Frente	26.51	0.59	9.14	0.20	SI
Mano Izquierda Abajo - Subir y Bajar	17.34	0.48	20.78	0.58	SI
Mano Izquierda Abajo - Subir y Bajar de Frente	26.91	0.59	11.08	0.24	SI
Mano Izquierda Abajo - Subir por completo y Bajar de Frente	29.60	0.59	5.11	0.10	NO
Saludo Mano Derecha	35.87	0.50	23.52	0.33	SI
Saludo Mano Izquierda	39.82	0.71	20.00	0.36	SI

Cuadro 8.44: Evaluación de 11 gestos con el cuerpo girado 45º con perfil Centrado y Rotado mediante Cambio de Base

En esta evaluación sólo se ha detectado un gesto erróneamente: *Mano Izquierda Abajo - Subir por completo y Bajar de Frente*. Este gesto se confunde con *Mano Izquierda Abajo - Subir y Bajar*, un gesto que es muy parecido. De hecho en esta ocasión el problema de la detección proviene de los límites del propio Kinect, ya que el esqueleto detectado por Kinect al levantar por completo la mano no la detecta bien. Al grabar el gesto girado, el esqueleto estaba más cerca del Kinect y la mano levantada por encima de la cabeza está fuera de los límites del Kinect, por lo que la mano queda detectada a la altura de la cabeza, en una posición muy parecida a la posición del gesto de *Mano Izquierda Abajo - Subir y Bajar*.

La Probabilidad acumulada de distancias de gesto detectados y la Probabilidad acumulada de distancias de gesto detectado y segundo nos dan una idea

del porcentaje de gestos que están por debajo de una determinada distancia:

En general observamos que las distancias de los gestos detectados son valores dentro de la media: ni muy grandes ni muy pequeños. En la primera gráfica, observando la línea roja, podemos ver que más del 50 % de los gestos tienen una distancia inferior a 26 metros.

Las distancias entre cada gesto detectado y el siguiente, el segundo gesto más parecido, tiene valores pequeños. Observando la línea azul, podemos ver que para más del 50 % de los gestos la distancia entre el gesto detectado y el segundo es menor de 10 metros, frente a una distancia media de los gestos detectados de 26 metros.

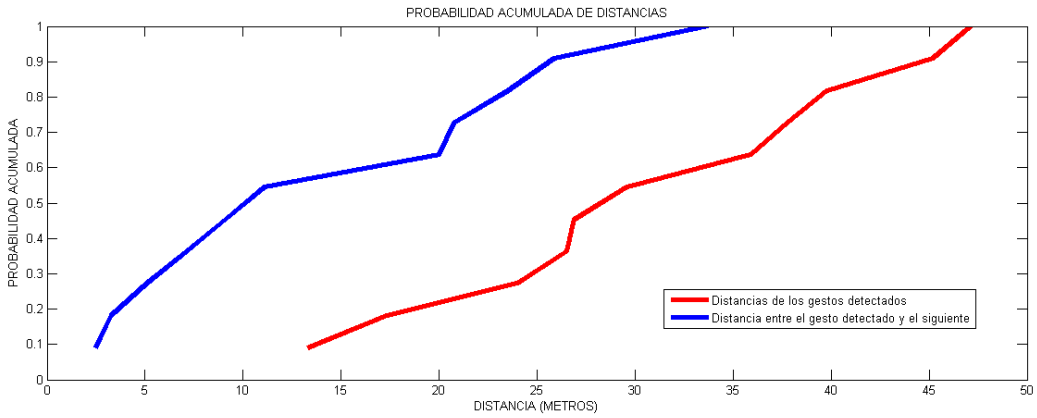


Figura 8.19: Probabilidad acumulada de distancias

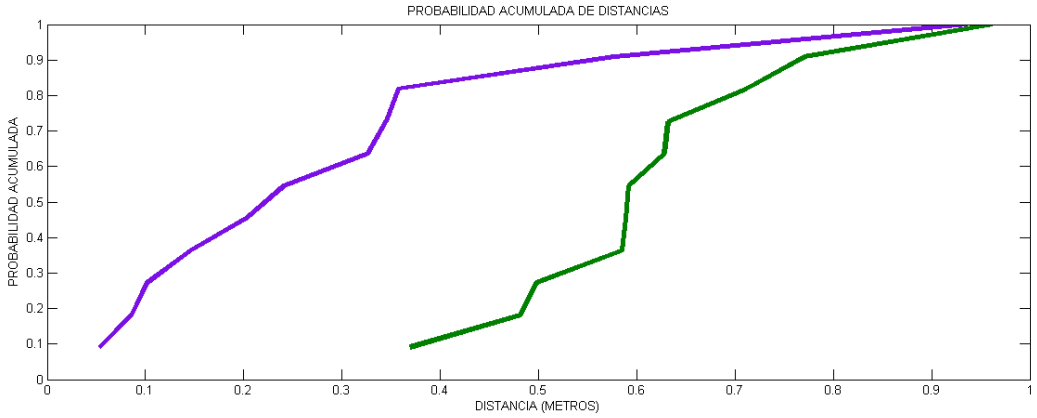


Figura 8.20: Probabilidad acumulada de distancias entre frames

Como vemos en la línea verde de la segunda gráfica, todas las distancias están por debajo de 1. Por tanto, los resultados de la matriz de confusión para $\mathbf{U_{DG}} = \mathbf{1}$ serán los mismos resultados que sin umbral.

Observando la línea morada, vemos que hay 4 gestos por debajo del 0,2: 3 gestos acertados y 1 fallado. Los resultados de la matriz de confusión para $\mathbf{U_{GDS}} = \mathbf{0.2}$ serán peores que los resultados sin umbral.

Observando los matrices de confusión, comprobamos que los resultados son iguales tanto para $\mathbf{U_{DG}} = \mathbf{1}$ como sin umbral. Los resultados para estos dos casos son bastante buenos: **acierto** $= \mathbf{91\,\%}$ y valor $\mathbf{F_1} = \mathbf{95\,\%}$.

$\mathbf{U_{DG}} = \mathbf{1}$

Umbral distancia deteccion de gestos

	GA	GF	
GDT	(TP) 10	(FP) 1	PRE 0.91
GDS	(FN) 0	(TN) 0	NPV -
F_1 **0.95**	REC 1	ESP 0	ACC **0.91**

$\mathbf{U_{GDS}} = \mathbf{0.2}$

Umbral distancia gesto detectado y segundo

	GA	GF	
GDT	(TP) 7	(FP) 0	PRE 1
GDS	(FN) 3	(TN) 1	NPV 0.25
F_1 **0.82**	REC 0.7	ESP 1	ACC **0.73**

$\mathbf{U_{DG}} = \mathbf{1}$ y $\mathbf{U_{GDS}} = \mathbf{0.2}$

	GA	GF	
GDT	(TP) 7	(FP) 0	PRE 1
GDS	(FN) 3	(TN) 1	NPV 0.25
F_1 **0.82**	REC 0.7	ESP 1	ACC **0.73**

Sin Umbral

	GA	GF	
GDT	(TP) 10	(FP) 1	PRE 0.9
GDS	(FN) 0	(TN) 0	NPV -
F_1 **0.95**	REC 1	ESP 0	ACC **0.91**

Tabla de Valores F-1 y Acierto para distintos Umbrales

U_{DG}[1]	U_{DS}[2]	F_1[3]	AC[4]
0.02	-	-	0.0909
0.04	-	-	0.0909
0.06	-	-	0.0909
0.08	-	-	0.0909
0.1	-	-	0.0909
0.12	-	-	0.0909
0.14	-	-	0.0909
0.16	-	-	0.0909
0.18	-	-	0.0909
0.2	-	-	0.0909
0.22	-	-	0.0909
0.24	-	-	0.0909
0.26	-	-	0.0909
0.28	-	-	0.0909
0.3	-	-	0.0909
0.32	-	-	0.0909
0.34	-	-	0.0909
0.36	-	-	0.0909
0.38	-	0.1818	0.1818
0.4	-	0.1818	0.1818
0.42	-	0.1818	0.1818
0.44	-	0.1818	0.1818
0.46	-	0.1818	0.1818
0.48	-	0.1818	0.1818
0.5	-	0.4615	0.3636
0.52	-	0.4615	0.3636
0.54	-	0.4615	0.3636
0.56	-	0.4615	0.3636
0.58	-	0.4615	0.3636
0.6	-	0.6250	0.4545
0.62	-	0.6250	0.4545
0.64	-	0.7778	0.6364
0.66	-	0.7778	0.6364
0.68	-	0.7778	0.6364
0.7	-	0.7778	0.6364
0.72	-	0.8421	0.7273
0.74	-	0.8421	0.7273
0.76	-	0.8421	0.7273
0.78	-	0.9000	0.8182
0.8	-	0.9000	0.8182
0.82	-	0.9000	0.8182
0.84	-	0.9000	0.8182
0.86	-	0.9000	0.8182
0.88	-	0.9000	0.8182
0.9	-	0.9000	0.8182
0.92	-	0.9000	0.8182
0.94	-	0.9000	0.8182
0.96	-	0.9000	0.8182
0.98	-	0.9524	0.9091
1	-	0.9524	0.9091

U_{DG}[1]	U_{DS}[2]	F_1[3]	AC[4]
-	0.02	0.9524	0.9091
-	0.04	0.9524	0.9091
-	0.06	0.9000	0.8182
-	0.08	0.9000	0.8182
-	0.1	0.8421	0.7273
-	0.12	0.8889	0.8182
-	0.14	0.8889	0.8182
-	0.16	0.8235	0.7273
-	0.18	0.8235	0.7273
-	0.2	0.8235	0.7273
-	0.22	0.7500	0.6364
-	0.24	0.7500	0.6364
-	0.26	0.6667	0.5455
-	0.28	0.6667	0.5455
-	0.3	0.6667	0.5455
-	0.32	0.6667	0.5455
-	0.34	0.5714	0.4545
-	0.36	0.3333	0.2727
-	0.38	0.3333	0.2727
-	0.4	0.3333	0.2727
-	0.42	0.3333	0.2727
-	0.44	0.3333	0.2727
-	0.46	0.3333	0.2727
-	0.48	0.3333	0.2727
-	0.5	0.3333	0.2727
-	0.52	0.3333	0.2727
-	0.54	0.3333	0.2727
-	0.56	0.3333	0.2727
-	0.58	0.1818	0.1818
-	0.6	0.1818	0.1818
-	0.62	0.1818	0.1818
-	0.64	0.1818	0.1818
-	0.66	0.1818	0.1818
-	0.68	0.1818	0.1818
-	0.7	0.1818	0.1818
-	0.72	0.1818	0.1818
-	0.74	0.1818	0.1818
-	0.76	0.1818	0.1818
-	0.78	0.1818	0.1818
-	0.8	0.1818	0.1818
-	0.82	0.1818	0.1818
-	0.84	0.1818	0.1818
-	0.86	0.1818	0.1818
-	0.88	0.1818	0.1818
-	0.9	0.1818	0.1818
-	0.92	0.1818	0.1818
-	0.94	-	0.0909
-	0.96	-	0.0909
-	0.98	-	0.0909
-	1	-	0.0909

U_{DG}[1]	U_{DS}[2]	F_1[3]	AC[4]
0.05	1	-	0.0909
0.05	0.9	-	0.0909
0.05	0.8	-	0.0909
0.1	0.95	-	0.0909
0.1	0.85	-	0.0909
0.1	0.75	-	0.0909
0.15	0.9	-	0.0909
0.15	0.8	-	0.0909
0.15	0.7	-	0.0909
0.2	0.85	-	0.0909
0.2	0.75	-	0.0909
0.2	0.65	-	0.0909
0.25	0.8	-	0.0909
0.25	0.7	-	0.0909
0.25	0.6	-	0.0909
0.3	0.75	-	0.0909
0.3	0.65	-	0.0909
0.3	0.55	-	0.0909
0.35	0.7	-	0.0909
0.35	0.6	-	0.0909
0.35	0.5	-	0.0909
0.4	0.65	0.1818	0.1818
0.4	0.55	0.1818	0.1818
0.4	0.45	0.1818	0.1818
0.45	0.6	0.1818	0.1818
0.45	0.5	0.1818	0.1818
0.45	0.4	0.1818	0.1818
0.5	0.55	0.3333	0.2727
0.5	0.45	0.3333	0.2727
0.5	0.35	0.3333	0.2727
0.55	0.5	0.3333	0.2727
0.55	0.4	0.3333	0.2727
0.55	0.3	0.4615	0.3636
0.6	0.45	0.3333	0.2727
0.6	0.35	0.3333	0.2727
0.6	0.25	0.4615	0.3636
0.65	0.4	0.3333	0.2727
0.65	0.3	0.5714	0.4545
0.65	0.2	0.7500	0.6364
0.7	0.35	0.3333	0.2727
0.7	0.25	0.5714	0.4545
0.7	0.15	0.7500	0.6364
0.75	0.3	0.6667	0.5455
0.75	0.2	0.8235	0.7273
0.75	0.1	0.7778	0.6364
0.8	0.25	0.6667	0.5455
0.8	0.15	0.8235	0.7273
0.8	0.05	0.9000	0.8182
0.85	0.2	0.8235	0.7273
0.85	0.1	0.8421	0.7273

1 - $\mathbf{U_{DG}}$ - Umbral de distancia del gesto detectado
2 - $\mathbf{U_{GDS}}$ - Umbral de distancia entre el gesto detectado y el segundo
3 - $\mathbf{F_1}$ - Valor F-1
4 - **AC** - Acierto

Evaluación de 16 gestos con 5 gestos inexistentes

El porcentaje de acierto es del 69 % (11 gestos acertados de 16) debido a que 5 gestos son inexistentes y se van a fallar seguro. Sin embargo, es interesante analizar el umbral de rechazo para ver en qué casos esos gestos van a ser descartados por no superar el umbral de rechazo, y pasar de Falsos Positivos (FP) a Verdaderos Negativos (TN).

GESTO	DISTANCIA GESTO DETECTADO	DISTANCIA GESTO DETECTADO / FRAMES	DISTANCIA ENTRE GESTO DETECTADO Y SEGUNDO	DISTANCIA ENTRE GESTO DETECTADO Y SEGUNDO / FRAMES	ACIERTO
Ambas Manos Abajo - Subir y Bajar	12.31	0.27	42.30	0.92	SI
Ambas Manos Abajo - Subir y Bajar de Frente	11.09	0.25	27.60	0.61	SI
Ambas Manos Abajo - Subir por completo y Bajar de Frente	25.99	0.50	7.02	0.13	SI
Mano Derecha Abajo - Subir y Bajar	6.87	0.17	36.98	0.92	SI
Mano Derecha Abajo - Subir y Bajar de Frente	16.23	0.34	15.23	0.32	SI
Mano Derecha Abajo - Subir por completo y Bajar de Frente	14.38	0.27	6.52	0.12	SI
Mano Izquierda Abajo - Subir y Bajar	8.71	0.21	26.81	0.65	SI
Mano Izquierda Abajo - Subir y Bajar de Frente	5.71	0.14	14.79	0.36	SI
Mano Izquierda Abajo - Subir por completo y Bajar de Frente	10.98	0.22	11.07	0.23	SI
Saludo Mano Derecha	19.07	0.29	34.61	0.53	SI
Saludo Mano Izquierda	21.05	0.35	22.68	0.38	SI
Mano Derecha Abajo - Subir y Bajar Cruzado	37.47	0.83	16.18	0.36	NO
Mano Derecha Abajo - Mover a Izquierda y a Derecha	53.69	1.28	21.00	0.50	NO
Mano Izquierda Abajo - Subir de Frente	39.36	1.64	19.80	0.83	NO
Mano Izquierda Abajo - Mover a Derecha	77.10	1.61	17.16	0.36	NO
Ambas Manos Arriba - Bajar de Frente	37.71	1.35	21.64	0.77	NO

Cuadro 8.50: Evaluación de 16 gestos con perfil Centrado y Rotado mediante Cambio de Base

En la primera gráfica podemos observamos que para un 60 % de los gestos la distancia de los gestos detectados es inferior a la distancia que separa el gesto detectado y el segundo más próximo. Sin embargo, para un 40 % de los gestos, la

distancia de los gestos detectados es superior.

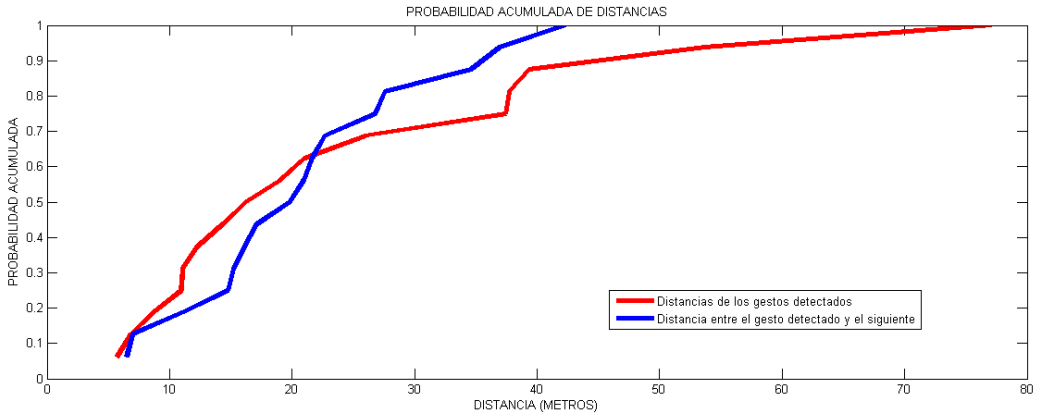


Figura 8.21: Probabilidad acumulada de distancias

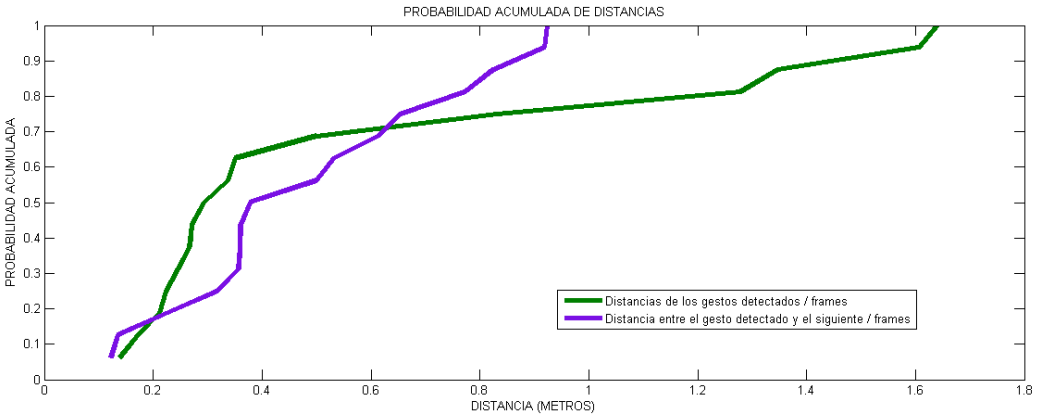


Figura 8.22: Probabilidad acumulada de distancias entre frames

Observando la línea morada en la segunda gráfica, vemos que sólo hay 2 gestos cuya distancia entre el gestos detectado y el segundo más próximo divido entre el número de frames sea menor de 0,2. Y estos dos gestos son gestos detectados correctamente, por lo que los resultados de la matriz de confusión serán peores que los resultados sin umbral.

Sin embargo, observamos que la distancia de los gestos detectados (línea verde) nos aporta una información muy útil. Todos los gestos fallados tienen una distancia de detección superior a 0,83, mientras que los gestos acertados tienen una distancia de detección inferior a 0,5. En este caso el umbral de rechazo

$\mathbf{U_{DG}} = 1$ funcionará perfectamente descartando solamente los gestos fallados. Sin embargo, todavía queda un gesto fallado que no ha sido descartado. El umbral de rechazo $\mathbf{U_{DG}}$ ideal en esta evaluación está entre 0,5 y 0,83.

Observando los matrices de confusión, comprobamos que los mejores resultados se obtienen con $\mathbf{U_{DG}} = \mathbf{1}$. En este caso, los resultados son bastante buenos: **acierto** $= \mathbf{94\,\%}$ y valor $\mathbf{F_1} = \mathbf{96\,\%}$. Con un umbral entre 0,5 y 0,83 los resultados serían del 100 %.

$\mathbf{U_{DG}} = \mathbf{1}$

Umbral distancia deteccion de gestos

	GA	GF	
GDT	(TP) 11	(FP) 1	PRE 0.92
GDS	(FN) 0	(TN) 4	NPV 1
F_1 **0.96**	REC 1	ESP 0.8	ACC **0.94**

$\mathbf{U_{GDS}} = \mathbf{0.2}$

Umbral distancia gesto detectado y segundo

	GA	GF	
GDT	(TP) 9	(FP) 5	PRE 0.64
GDS	(FN) 2	(TN) 0	NPV 0
F_1 **0.72**	REC 0.82	ESP 0	ACC **0.56**

$\mathbf{U_{DG}} = \mathbf{1}$ y $\mathbf{U_{GDS}} = \mathbf{0.2}$

	GA	GF	
GDT	(TP) 9	(FP) 1	PRE 0.9
GDS	(FN) 2	(TN) 4	NPV 0.67
F_1 **0.86**	REC 0.82	ESP 0.8	ACC **0.81**

Sin Umbral

	GA	GF	
GDT	(TP) 11	(FP) 5	PRE 0.69
GDS	(FN) 0	(TN) 0	NPV -
F_1 **0.81**	REC 1	ESP 0	ACC **0.69**

Tabla de Valores F-1 y Acierto para distintos Umbrales

U_{DG}[1]	U_{DS}[2]	F_1[3]	AC[4]
0.02	-	-	0.3125
0.04	-	-	0.3125
0.06	-	-	0.3125
0.08	-	-	0.3125
0.1	-	-	0.3125
0.12	-	-	0.3125
0.14	-	0.1667	0.3750
0.16	-	0.1667	0.3750
0.18	-	0.3077	0.4375
0.2	-	0.3077	0.4375
0.22	-	0.4286	0.5000
0.24	-	0.5333	0.5625
0.26	-	0.6250	0.6250
0.28	-	0.7778	0.7500
0.3	-	0.8421	0.8125
0.32	-	0.8421	0.8125
0.34	-	0.9000	0.8750
0.36	-	0.9524	0.9375
0.38	-	0.9524	0.9375
0.4	-	0.9524	0.9375
0.42	-	0.9524	0.9375
0.44	-	0.9524	0.9375
0.46	-	0.9524	0.9375
0.48	-	0.9524	0.9375
0.5	-	1.0000	1.0000
0.52	-	1.0000	1.0000
0.54	-	1.0000	1.0000
0.56	-	1.0000	1.0000
0.58	-	1.0000	1.0000
0.6	-	1.0000	1.0000
0.62	-	1.0000	1.0000
0.64	-	1.0000	1.0000
0.66	-	1.0000	1.0000
0.68	-	1.0000	1.0000
0.7	-	1.0000	1.0000
0.72	-	1.0000	1.0000
0.74	-	1.0000	1.0000
0.76	-	1.0000	1.0000
0.78	-	1.0000	1.0000
0.8	-	1.0000	1.0000
0.82	-	1.0000	1.0000
0.84	-	0.9565	0.9375
0.86	-	0.9565	0.9375
0.88	-	0.9565	0.9375
0.9	-	0.9565	0.9375
0.92	-	0.9565	0.9375
0.94	-	0.9565	0.9375
0.96	-	0.9565	0.9375
0.98	-	0.9565	0.9375
1	-	0.9565	0.9375

U_{DG}[1]	U_{DS}[2]	F_1[3]	AC[4]
-	0.02	0.8148	0.6875
-	0.04	0.8148	0.6875
-	0.06	0.8148	0.6875
-	0.08	0.8148	0.6875
-	0.1	0.8148	0.6875
-	0.12	0.8148	0.6875
-	0.14	0.7200	0.5625
-	0.16	0.7200	0.5625
-	0.18	0.7200	0.5625
-	0.2	0.7200	0.5625
-	0.22	0.7200	0.5625
-	0.24	0.6667	0.5000
-	0.26	0.6667	0.5000
-	0.28	0.6667	0.5000
-	0.3	0.6667	0.5000
-	0.32	0.6087	0.4375
-	0.34	0.6087	0.4375
-	0.36	0.6667	0.5625
-	0.38	0.5263	0.4375
-	0.4	0.5263	0.4375
-	0.42	0.5263	0.4375
-	0.44	0.5263	0.4375
-	0.46	0.5263	0.4375
-	0.48	0.5263	0.4375
-	0.5	0.5556	0.5000
-	0.52	0.5556	0.5000
-	0.54	0.4706	0.4375
-	0.56	0.4706	0.4375
-	0.58	0.4706	0.4375
-	0.6	0.4706	0.4375
-	0.62	0.3750	0.3750
-	0.64	0.3750	0.3750
-	0.66	0.2667	0.3125
-	0.68	0.2667	0.3125
-	0.7	0.2667	0.3125
-	0.72	0.2667	0.3125
-	0.74	0.2667	0.3125
-	0.76	0.2667	0.3125
-	0.78	0.2857	0.3750
-	0.8	0.2857	0.3750
-	0.82	0.2857	0.3750
-	0.84	0.3077	0.4375
-	0.86	0.3077	0.4375
-	0.88	0.3077	0.4375
-	0.9	0.3077	0.4375
-	0.92	0.1667	0.3750
-	0.94	-	0.3125
-	0.96	-	0.3125
-	0.98	-	0.3125
-	1	-	0.3125

U_{DG}[1]	U_{DS}[2]	F_1[3]	AC[4]
0.05	1	-	0.3125
0.05	0.9	-	0.3125
0.05	0.8	-	0.3125
0.1	0.95	-	0.3125
0.1	0.85	-	0.3125
0.1	0.75	-	0.3125
0.15	0.9	-	0.3125
0.15	0.8	-	0.3125
0.15	0.7	-	0.3125
0.2	0.85	0.1667	0.3750
0.2	0.75	0.1667	0.3750
0.2	0.65	0.1667	0.3750
0.25	0.8	0.1667	0.3750
0.25	0.7	0.1667	0.3750
0.25	0.6	0.4286	0.5000
0.3	0.75	0.3077	0.4375
0.3	0.65	0.4286	0.5000
0.3	0.55	0.5333	0.5625
0.35	0.7	0.3077	0.4375
0.35	0.6	0.5333	0.5625
0.35	0.5	0.6250	0.6250
0.4	0.65	0.4286	0.5000
0.4	0.55	0.5333	0.5625
0.4	0.45	0.6250	0.6250
0.45	0.6	0.5333	0.5625
0.45	0.5	0.6250	0.6250
0.45	0.4	0.6250	0.6250
0.5	0.55	0.5333	0.5625
0.5	0.45	0.6250	0.6250
0.5	0.35	0.7778	0.7500
0.55	0.5	0.6250	0.6250
0.55	0.4	0.6250	0.6250
0.55	0.3	0.8421	0.8125
0.6	0.45	0.6250	0.6250
0.6	0.35	0.7778	0.7500
0.6	0.25	0.8421	0.8125
0.65	0.4	0.6250	0.6250
0.65	0.3	0.8421	0.8125
0.65	0.2	0.9000	0.8750
0.7	0.35	0.7778	0.7500
0.7	0.25	0.8421	0.8125
0.7	0.15	0.9000	0.8750
0.75	0.3	0.8421	0.8125
0.75	0.2	0.9000	0.8750
0.75	0.1	1.0000	1.0000
0.8	0.25	0.8421	0.8125
0.8	0.15	0.9000	0.8750
0.8	0.05	1.0000	1.0000
0.85	0.2	0.8571	0.8125
0.85	0.1	0.9565	0.9375

1 - $\mathbf{U_{DG}}$ - Umbral de distancia del gesto detectado
2 - $\mathbf{U_{GDS}}$ - Umbral de distancia entre el gesto detectado y el segundo
3 - $\mathbf{F_1}$ - Valor F-1
4 - **AC** - Acierto

Evaluaciones con el esqueleto centrado y rotado mediante giro

Al igual que ocurría con el perfil anterior, este perfil de configuración nos permitirá comparar los gestos como si se hubieran realizado todos de frente y en la misma posición.

Como vemos en la siguiente tabla, y también veremos con más detalle a continuación en el análisis de cada evaluación, este preprocesamiento va a eliminar las dos principales causas de que se detecten gestos erróneamente con el perfil sin preprocesamiento: gestos realizados en distintas posiciones y gestos realizados con el cuerpo girado.

Evaluación de 11 gestos

Como vemos en la siguiente tabla, este perfil obtiene unos resultados muy similares al perfil de *centrado y rotado mediante cambio de base*: porcentaje de acierto del 100 % y unas distancias de detección de gestos normales.

GESTO	DISTANCIA GESTO DETECTADO	DISTANCIA GESTO DETECTADO / FRAMES	DISTANCIA ENTRE GESTO DETECTADO Y SEGUNDO	DISTANCIA ENTRE GESTO DETECTADO Y SEGUNDO / FRAMES	ACIERTO
Ambas Manos Abajo - Subir y Bajar	11.72	0.26	40.66	0.90	SI
Ambas Manos Abajo - Subir y Bajar de Frente	12.47	0.27	44.43	0.97	SI
Ambas Manos Abajo - Subir por completo y Bajar de Frente	27.97	0.45	32.88	0.53	SI
Mano Derecha Abajo - Subir y Bajar	11.92	0.25	37.25	0.79	SI
Mano Derecha Abajo - Subir y Bajar de Frente	18.57	0.37	30.08	0.60	SI
Mano Derecha Abajo - Subir por completo y Bajar de Frente	22.13	0.36	32.09	0.53	SI
Mano Izquierda Abajo - Subir y Bajar	12.29	0.30	36.45	0.89	SI
Mano Izquierda Abajo - Subir y Bajar de Frente	9.19	0.23	32.15	0.80	SI
Mano Izquierda Abajo - Subir por completo y Bajar de Frente	16.43	0.29	26.61	0.48	SI
Saludo Mano Derecha	28.84	0.41	70.35	0.99	SI
Saludo Mano Izquierda	36.23	0.50	50.95	0.71	SI

Cuadro 8.56: Evaluación de 11 gestos con perfil Centrado y Rotado mediante Giro eje Y

Más del 50 % de los gestos detectado tiene una distancia inferior a 15 metros, mucho menor que la diferencia de distancia con el gesto más próximo, que es superior a 35 metros para más del 50 % de los gesto. Es decir, la diferencia de distancia entre el gestos detectado y el siguiente es más del doble que la distancia de detección.

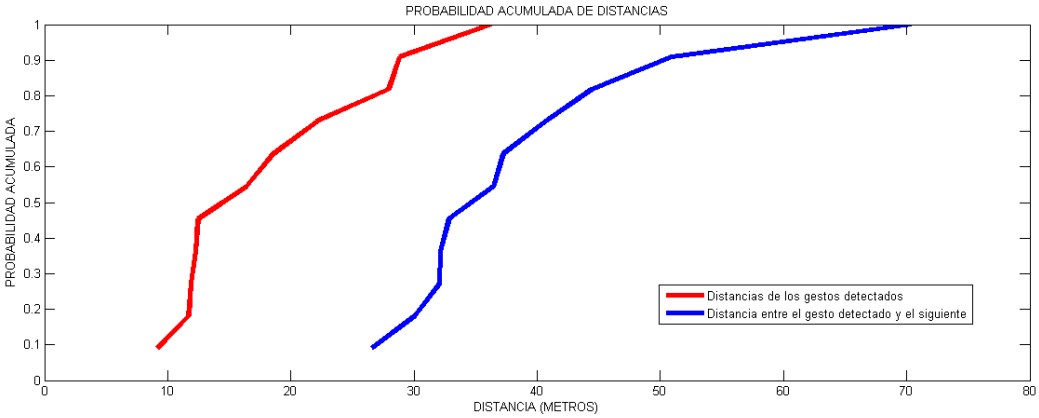


Figura 8.23: Probabilidad acumulada de distancias

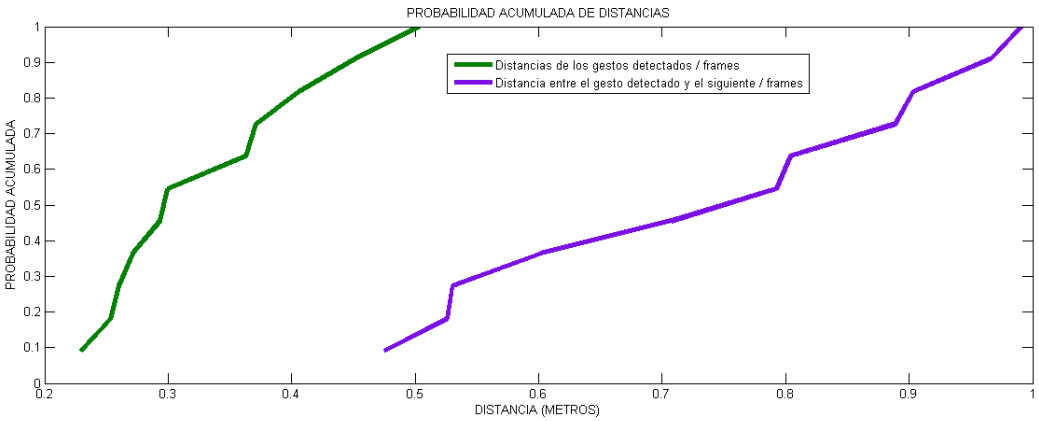


Figura 8.24: Probabilidad acumulada de distancias entre frames

En esta evaluación todos los gestos han sido detectados correctamente, con unas distancias de detección de gestos bajas. Como vemos en la línea verde de la segunda gráfica, la distancia de los gestos detectados está por debajo de 0,5 ($\mathbf{U_{DG}} = 1 > 0{,}5$. Además, la distancia entre el gesto detectado y el segundo más próximo está por encima de 0,23 ($\mathbf{U_{GDS}} = 0{,}2 < 0{,}23$. Tanto para $\mathbf{U_{DG}} = 1$ como

para $\mathbf{U_{GDS}} = 0{,}2$ no se va a descartar ningún gesto, por lo que los resultados de las matrices de confusión serán los mismos que sin umbral de rechazo.

Para todos los casos los resultados son muy buenos: **acierto** $= \mathbf{100\,\%}$ y valor $\mathbf{F_1} = \mathbf{100\,\%}$.

$\mathbf{U_{DG}} = \mathbf{1}$

Umbral distancia deteccion de gestos

	GA	GF	
GDT	(TP) 11	(FP) 0	PRE 1
GDS	(FN) 0	(TN) 0	NPV -
F_1 **1**	REC 1	ESP -	ACC **1**

$\mathbf{U_{GDS}} = \mathbf{0.2}$

Umbral distancia gesto detectado y segundo

	GA	GF	
GDT	(TP) 11	(FP) 0	PRE 1
GDS	(FN) 0	(TN) 0	NPV -
F_1 **1**	REC 1	ESP -	ACC **1**

$\mathbf{U_{DG}} = \mathbf{0.6}$ y $\mathbf{U_{GDS}} = \mathbf{0.2}$

	GA	GF	
GDT	(TP) 11	(FP) 0	PRE 1
GDS	(FN) 0	(TN) 0	NPV -
F_1 **1**	REC 1	ESP -	ACC **1**

Sin Umbral

	GA	GF	
GDT	(TP) 11	(FP) 0	PRE 1
GDS	(FN) 0	(TN) 0	NPV -
F_1 **1**	REC 1	ESP -	ACC **1**

Tabla de Valores F-1 y Acierto para distintos Umbrales

U_{DG}[1]	U_{DS}[2]	F_1[3]	AC[4]
0.02	-	-	0.0000
0.04	-	-	0.0000
0.06	-	-	0.0000
0.08	-	-	0.0000
0.1	-	-	0.0000
0.12	-	-	0.0000
0.14	-	-	0.0000
0.16	-	-	0.0000
0.18	-	-	0.0000
0.2	-	-	0.0000
0.22	-	-	0.0000
0.24	-	0.1667	0.0909
0.26	-	0.3077	0.1818
0.28	-	0.5333	0.3636
0.3	-	0.7059	0.5455
0.32	-	0.7059	0.5455
0.34	-	0.7059	0.5455
0.36	-	0.7059	0.5455
0.38	-	0.8421	0.7273
0.4	-	0.8421	0.7273
0.42	-	0.9000	0.8182
0.44	-	0.9000	0.8182
0.46	-	0.9524	0.9091
0.48	-	0.9524	0.9091
0.5	-	0.9524	0.9091
0.52	-	1.0000	1.0000
0.54	-	1.0000	1.0000
0.56	-	1.0000	1.0000
0.58	-	1.0000	1.0000
0.6	-	1.0000	1.0000
0.62	-	1.0000	1.0000
0.64	-	1.0000	1.0000
0.66	-	1.0000	1.0000
0.68	-	1.0000	1.0000
0.7	-	1.0000	1.0000
0.72	-	1.0000	1.0000
0.74	-	1.0000	1.0000
0.76	-	1.0000	1.0000
0.78	-	1.0000	1.0000
0.8	-	1.0000	1.0000
0.82	-	1.0000	1.0000
0.84	-	1.0000	1.0000
0.86	-	1.0000	1.0000
0.88	-	1.0000	1.0000
0.9	-	1.0000	1.0000
0.92	-	1.0000	1.0000
0.94	-	1.0000	1.0000
0.96	-	1.0000	1.0000
0.98	-	1.0000	1.0000
1	-	1.0000	1.0000

U_{DG}[1]	U_{DS}[2]	F_1[3]	AC[4]
-	0.02	1.0000	1.0000
-	0.04	1.0000	1.0000
-	0.06	1.0000	1.0000
-	0.08	1.0000	1.0000
-	0.1	1.0000	1.0000
-	0.12	1.0000	1.0000
-	0.14	1.0000	1.0000
-	0.16	1.0000	1.0000
-	0.18	1.0000	1.0000
-	0.2	1.0000	1.0000
-	0.22	1.0000	1.0000
-	0.24	1.0000	1.0000
-	0.26	1.0000	1.0000
-	0.28	1.0000	1.0000
-	0.3	1.0000	1.0000
-	0.32	1.0000	1.0000
-	0.34	1.0000	1.0000
-	0.36	1.0000	1.0000
-	0.38	1.0000	1.0000
-	0.4	1.0000	1.0000
-	0.42	1.0000	1.0000
-	0.44	1.0000	1.0000
-	0.46	1.0000	1.0000
-	0.48	0.9524	0.9091
-	0.5	0.9524	0.9091
-	0.52	0.9524	0.9091
-	0.54	0.8421	0.7273
-	0.56	0.8421	0.7273
-	0.58	0.8421	0.7273
-	0.6	0.8421	0.7273
-	0.62	0.7778	0.6364
-	0.64	0.7778	0.6364
-	0.66	0.7778	0.6364
-	0.68	0.7778	0.6364
-	0.7	0.7778	0.6364
-	0.72	0.7059	0.5455
-	0.74	0.7059	0.5455
-	0.76	0.7059	0.5455
-	0.78	0.7059	0.5455
-	0.8	0.6250	0.4545
-	0.82	0.5333	0.3636
-	0.84	0.5333	0.3636
-	0.86	0.5333	0.3636
-	0.88	0.5333	0.3636
-	0.9	0.4286	0.2727
-	0.92	0.3077	0.1818
-	0.94	0.3077	0.1818
-	0.96	0.3077	0.1818
-	0.98	0.1667	0.0909
-	1	-	0.0000

U_{DG}[1]	U_{DS}[2]	F_1[3]	AC[4]
0.05	1	-	0.0000
0.05	0.9	-	0.0000
0.05	0.8	-	0.0000
0.1	0.95	-	0.0000
0.1	0.85	-	0.0000
0.1	0.75	-	0.0000
0.15	0.9	-	0.0000
0.15	0.8	-	0.0000
0.15	0.7	-	0.0000
0.2	0.85	-	0.0000
0.2	0.75	-	0.0000
0.2	0.65	-	0.0000
0.25	0.8	0.1667	0.0909
0.25	0.7	0.1667	0.0909
0.25	0.6	0.1667	0.0909
0.3	0.75	0.6250	0.4545
0.3	0.65	0.6250	0.4545
0.3	0.55	0.6250	0.4545
0.35	0.7	0.6250	0.4545
0.35	0.6	0.6250	0.4545
0.35	0.5	0.6250	0.4545
0.4	0.65	0.6250	0.4545
0.4	0.55	0.7059	0.5455
0.4	0.45	0.8421	0.7273
0.45	0.6	0.7778	0.6364
0.45	0.5	0.8421	0.7273
0.45	0.4	0.9000	0.8182
0.5	0.55	0.7778	0.6364
0.5	0.45	0.9524	0.9091
0.5	0.35	0.9524	0.9091
0.55	0.5	0.9524	0.9091
0.55	0.4	1.0000	1.0000
0.55	0.3	1.0000	1.0000
0.6	0.45	1.0000	1.0000
0.6	0.35	1.0000	1.0000
0.6	0.25	1.0000	1.0000
0.65	0.4	1.0000	1.0000
0.65	0.3	1.0000	1.0000
0.65	0.2	1.0000	1.0000
0.7	0.35	1.0000	1.0000
0.7	0.25	1.0000	1.0000
0.7	0.15	1.0000	1.0000
0.75	0.3	1.0000	1.0000
0.75	0.2	1.0000	1.0000
0.75	0.1	1.0000	1.0000
0.8	0.25	1.0000	1.0000
0.8	0.15	1.0000	1.0000
0.8	0.05	1.0000	1.0000
0.85	0.2	1.0000	1.0000
0.85	0.1	1.0000	1.0000

1 - $\mathbf{U_{DG}}$ - Umbral de distancia del gesto detectado
2 - $\mathbf{U_{GDS}}$ - Umbral de distancia entre el gesto detectado y el segundo
3 - $\mathbf{F_1}$ - Valor F-1
4 - **AC** - Acierto

Evaluación de 11 gestos con el cuerpo girado 45º

Esta evaluación tiene un porcentaje de acierto del 64 %, bastante inferior al porcentaje de acierto del perfil de *centrado y rotado mediante cambio de base*, que era del 91 %.

GESTO	DISTANCIA GESTO DETECTADO	DISTANCIA GESTO DETECTADO / FRAMES	DISTANCIA ENTRE GESTO DETECTADO Y SEGUNDO	DISTANCIA ENTRE GESTO DETECTADO Y SEGUNDO / FRAMES	ACIERTO
Ambas Manos Abajo - Subir y Bajar	52.42	0.70	39.45	0.53	SI
Ambas Manos Abajo - Subir y Bajar de Frente	57.54	1.22	10.74	0.23	NO
Ambas Manos Abajo - Subir por completo y Bajar de Frente	53.80	1.10	17.94	0.37	SI
Mano Derecha Abajo - Subir y Bajar	19.25	0.53	29.99	0.83	SI
Mano Derecha Abajo - Subir y Bajar de Frente	28.76	0.76	22.42	0.59	NO
Mano Derecha Abajo - Subir por completo y Bajar de Frente	40.16	0.89	14.52	0.32	NO
Mano Izquierda Abajo - Subir y Bajar	32.26	0.90	8.73	0.24	SI
Mano Izquierda Abajo - Subir y Bajar de Frente	30.62	0.67	16.86	0.37	SI
Mano Izquierda Abajo - Subir por completo y Bajar de Frente	33.07	0.66	8.36	0.17	NO
Saludo Mano Derecha	55.71	0.77	27.97	0.39	SI
Saludo Mano Izquierda	47.34	0.85	17.75	0.32	SI

Cuadro 8.62: Evaluación de 11 gestos con el cuerpo girado 45º con perfil Centrado y Rotado mediante Giro eje Y

Si nos fijamos en los gestos que han fallado, son los **mismo gestos** que en el perfil de *centrado y rotado mediante cambio de base* tenían una distancia inferior a 0,2. En este caso, esos gestos han sido detectados incorrectamente. Sin embargo, ahora sólo uno de ellos tiene una distancia por debajo de 0,2. El resto tiene una distancia superior a 0,2.

En general observamos que las distancias de los gestos detectados son valores dentro de la media, aunque algo superiores al perfil de *centrado y rotado mediante cambio de base*. En la primera gráfica podemos observar que más del 50 % de los gestos tiene una distancia inferior a 35 metros.

Las distancias entre cada gesto detectado y el siguiente, el segundo gesto más parecido, tiene valores pequeños. En la segunda gráfica podemos observar que

para más del 50 % de los gestos la distancia entre el gesto detectado y el segundo es menor de 18 metros, frente a una distancia media de los gestos detectados de 35 metros.

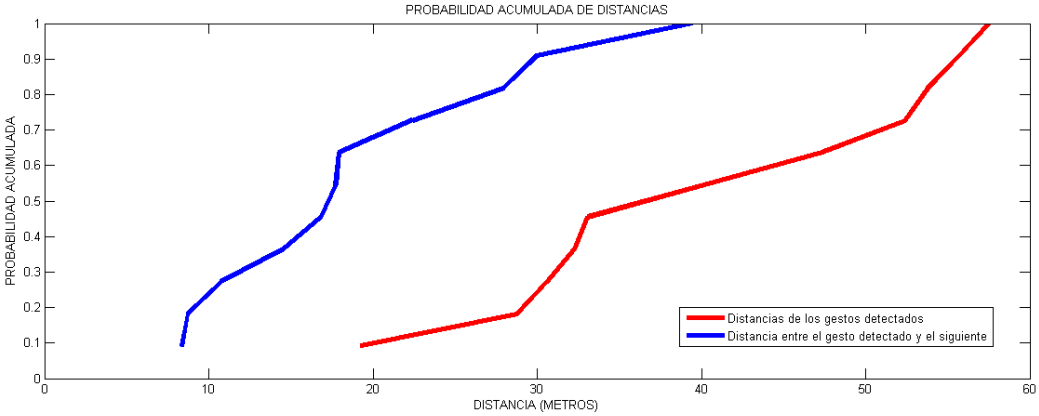


Figura 8.25: Probabilidad acumulada de distancias

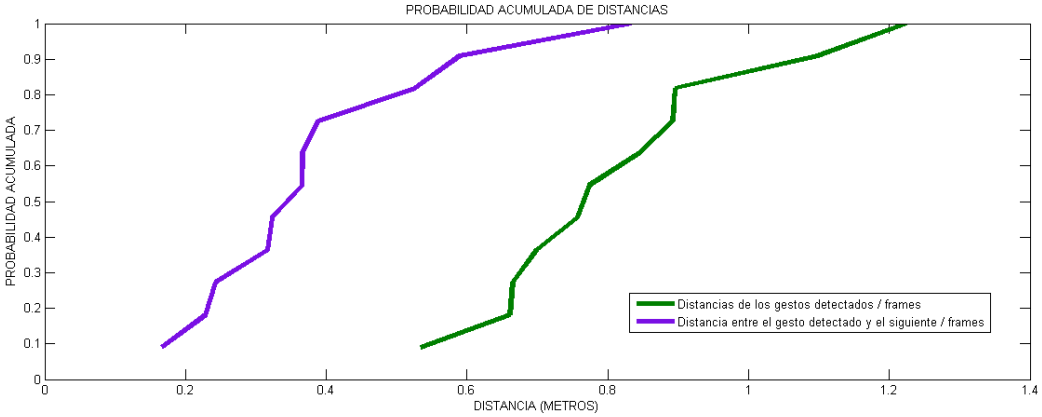


Figura 8.26: Probabilidad acumulada de distancias entre frames

Como vemos en la línea verde de la segunda gráfica, hay 2 gestos por encima del umbral $\mathbf{U_{DG}} = \mathbf{1}$: un gesto acertado y otro fallado. Por tanto, los resultados de la matriz de confusión para $\mathbf{U_{DG}} = \mathbf{1}$ serán prácticamente los mismos que los resultados sin umbral (el **acierto** será igual, y el valor $\mathbf{F}_1$ será ligeramente superior sin umbral).

Observando la línea morada, vemos que sólo hay 1 gestos por debajo del 0,2, y es un gesto fallado. Por tanto, los resultados de la matriz de confusión para $\mathbf{U_{GDS}} = \mathbf{0.2}$ serán mejores que los resultados sin umbral.

Observando los matrices de confusión, comprobamos que los mejores resultados se obtienen para $\mathbf{U_{GDS}} = \mathbf{0.2}$: **acierto** $= \mathbf{73\,\%}$ y valor $\mathbf{F_1} = \mathbf{82\,\%}$. Sin embargo, estos resultados son peores que los obtenidos con el perfil de *centrado y rotado mediante cambio de base.*

$\mathbf{U_{DG}} = \mathbf{1}$

Umbral distancia deteccion de gestos

	GA	GF	
GDT	(TP) 6	(FP) 3	PRE 0.67
GDS	(FN) 1	(TN) 1	NPV 0.5
F_1 **0.75**	REC 0.86	ESP 0.25	ACC **0.64**

$\mathbf{U_{GDS}} = \mathbf{0.2}$

Umbral distancia gesto detectado y segundo

	GA	GF	
GDT	(TP) 7	(FP) 3	PRE 0.7
GDS	(FN) 0	(TN) 1	NPV 1
F_1 **0.82**	REC 1	ESP 0.25	ACC **0.73**

$\mathbf{U_{DG}} = \mathbf{1}$ y $\mathbf{U_{GDS}} = \mathbf{0.2}$

	GA	GF	
GDT	(TP) 6	(FP) 2	PRE 0.75
GDS	(FN) 1	(TN) 2	NPV 0.67
F_1 **0.8**	REC 0.86	ESP 0.5	ACC **0.73**

Sin Umbral

	GA	GF	
GDT	(TP) 7	(FP) 4	PRE 0.64
GDS	(FN) 0	(TN) 0	NPV -
F_1 **0.78**	REC 1	ESP 0	ACC **0.64**

Tabla de Valores F-1 y Acierto para distintos Umbrales

U_{DG}[1]	U_{DS}[2]	F_1[3]	AC[4]
0.02	-	-	0.3636
0.04	-	-	0.3636
0.06	-	-	0.3636
0.08	-	-	0.3636
0.1	-	-	0.3636
0.12	-	-	0.3636
0.14	-	-	0.3636
0.16	-	-	0.3636
0.18	-	-	0.3636
0.2	-	-	0.3636
0.22	-	-	0.3636
0.24	-	-	0.3636
0.26	-	-	0.3636
0.28	-	-	0.3636
0.3	-	-	0.3636
0.32	-	-	0.3636
0.34	-	-	0.3636
0.36	-	-	0.3636
0.38	-	-	0.3636
0.4	-	-	0.3636
0.42	-	-	0.3636
0.44	-	-	0.3636
0.46	-	-	0.3636
0.48	-	-	0.3636
0.5	-	-	0.3636
0.52	-	-	0.3636
0.54	-	0.2500	0.4545
0.56	-	0.2500	0.4545
0.58	-	0.2500	0.4545
0.6	-	0.2500	0.4545
0.62	-	0.2500	0.4545
0.64	-	0.2500	0.4545
0.66	-	0.2500	0.4545
0.68	-	0.4000	0.4545
0.7	-	0.5455	0.5455
0.72	-	0.5455	0.5455
0.74	-	0.5455	0.5455
0.76	-	0.5000	0.4545
0.78	-	0.6154	0.5455
0.8	-	0.6154	0.5455
0.82	-	0.6154	0.5455
0.84	-	0.6154	0.5455
0.86	-	0.7143	0.6364
0.88	-	0.7143	0.6364
0.9	-	0.7500	0.6364
0.92	-	0.7500	0.6364
0.94	-	0.7500	0.6364
0.96	-	0.7500	0.6364
0.98	-	0.7500	0.6364
1	-	0.7500	0.6364

U_{DG}[1]	U_{DS}[2]	F_1[3]	AC[4]
-	0.02	0.7778	0.6364
-	0.04	0.7778	0.6364
-	0.06	0.7778	0.6364
-	0.08	0.7778	0.6364
-	0.1	0.7778	0.6364
-	0.12	0.7778	0.6364
-	0.14	0.7778	0.6364
-	0.16	0.7778	0.6364
-	0.18	0.8235	0.7273
-	0.2	0.8235	0.7273
-	0.22	0.8235	0.7273
-	0.24	0.8750	0.8182
-	0.26	0.8000	0.7273
-	0.28	0.8000	0.7273
-	0.3	0.8000	0.7273
-	0.32	0.7143	0.6364
-	0.34	0.7692	0.7273
-	0.36	0.7692	0.7273
-	0.38	0.5455	0.5455
-	0.4	0.4000	0.4545
-	0.42	0.4000	0.4545
-	0.44	0.4000	0.4545
-	0.46	0.4000	0.4545
-	0.48	0.4000	0.4545
-	0.5	0.4000	0.4545
-	0.52	0.4000	0.4545
-	0.54	0.2222	0.3636
-	0.56	0.2222	0.3636
-	0.58	0.2222	0.3636
-	0.6	0.2500	0.4545
-	0.62	0.2500	0.4545
-	0.64	0.2500	0.4545
-	0.66	0.2500	0.4545
-	0.68	0.2500	0.4545
-	0.7	0.2500	0.4545
-	0.72	0.2500	0.4545
-	0.74	0.2500	0.4545
-	0.76	0.2500	0.4545
-	0.78	0.2500	0.4545
-	0.8	0.2500	0.4545
-	0.82	0.2500	0.4545
-	0.84	-	0.3636
-	0.86	-	0.3636
-	0.88	-	0.3636
-	0.9	-	0.3636
-	0.92	-	0.3636
-	0.94	-	0.3636
-	0.96	-	0.3636
-	0.98	-	0.3636
-	1	-	0.3636

U_{DG}[1]	U_{DS}[2]	F_1[3]	AC[4]
0.05	1	-	0.3636
0.05	0.9	-	0.3636
0.05	0.8	-	0.3636
0.1	0.95	-	0.3636
0.1	0.85	-	0.3636
0.1	0.75	-	0.3636
0.15	0.9	-	0.3636
0.15	0.8	-	0.3636
0.15	0.7	-	0.3636
0.2	0.85	-	0.3636
0.2	0.75	-	0.3636
0.2	0.65	-	0.3636
0.25	0.8	-	0.3636
0.25	0.7	-	0.3636
0.25	0.6	-	0.3636
0.3	0.75	-	0.3636
0.3	0.65	-	0.3636
0.3	0.55	-	0.3636
0.35	0.7	-	0.3636
0.35	0.6	-	0.3636
0.35	0.5	-	0.3636
0.4	0.65	-	0.3636
0.4	0.55	-	0.3636
0.4	0.45	-	0.3636
0.45	0.6	-	0.3636
0.45	0.5	-	0.3636
0.45	0.4	-	0.3636
0.5	0.55	-	0.3636
0.5	0.45	-	0.3636
0.5	0.35	-	0.3636
0.55	0.5	0.2500	0.4545
0.55	0.4	0.2500	0.4545
0.55	0.3	0.2500	0.4545
0.6	0.45	0.2500	0.4545
0.6	0.35	0.2500	0.4545
0.6	0.25	0.2500	0.4545
0.65	0.4	0.2500	0.4545
0.65	0.3	0.2500	0.4545
0.65	0.2	0.2500	0.4545
0.7	0.35	0.6000	0.6364
0.7	0.25	0.6000	0.6364
0.7	0.15	0.5455	0.5455
0.75	0.3	0.6000	0.6364
0.75	0.2	0.6000	0.6364
0.75	0.1	0.5455	0.5455
0.8	0.25	0.6667	0.6364
0.8	0.15	0.6154	0.5455
0.8	0.05	0.6154	0.5455
0.85	0.2	0.7692	0.7273
0.85	0.1	0.7143	0.6364

1 - $\mathbf{U_{DG}}$ - Umbral de distancia del gesto detectado
2 - $\mathbf{U_{GDS}}$ - Umbral de distancia entre el gesto detectado y el segundo
3 - $\mathbf{F_1}$ - Valor F-1
4 - **AC** - Acierto

Evaluación de 16 gestos con 5 gestos inexistentes

El porcentaje de acierto es del 69 % (11 gestos acertados de 16) debido a que 5 gestos son inexistentes y se van a fallar seguro. Sin embargo, es interesante analizar el umbral de rechazo para ver en qué casos esos gestos van a ser descartados por no superar el umbral de rechazo, y pasar de Falsos Positivos (FP) a Verdaderos Negativos (TN).

GESTO	DISTANCIA GESTO DETECTADO	DISTANCIA GESTO DETECTADO / FRAMES	DISTANCIA ENTRE GESTO DETECTADO Y SEGUNDO	DISTANCIA ENTRE GESTO DETECTADO Y SEGUNDO / FRAMES	ACIERTO
Ambas Manos Abajo - Subir y Bajar	13.70	0.30	40.91	0.89	SI
Ambas Manos Abajo - Subir y Bajar de Frente	10.94	0.24	27.75	0.62	SI
Ambas Manos Abajo - Subir por completo y Bajar de Frente	25.73	0.49	7.28	0.14	SI
Mano Derecha Abajo - Subir y Bajar	8.95	0.22	34.89	0.87	SI
Mano Derecha Abajo - Subir y Bajar de Frente	16.39	0.34	15.07	0.31	SI
Mano Derecha Abajo - Subir por completo y Bajar de Frente	16.37	0.31	4.52	0.09	SI
Mano Izquierda Abajo - Subir y Bajar	9.05	0.22	26.47	0.65	SI
Mano Izquierda Abajo - Subir y Bajar de Frente	5.97	0.15	14.52	0.35	SI
Mano Izquierda Abajo - Subir por completo y Bajar de Frente	11.29	0.23	10.76	0.22	SI
Saludo Mano Derecha	19.64	0.30	34.05	0.52	SI
Saludo Mano Izquierda	21.39	0.36	22.34	0.37	SI
Mano Derecha Abajo - Subir y Bajar Cruzado	40.61	0.90	13.03	0.29	NO
Mano Derecha Abajo - Mover a Izquierda y a Derecha	65.32	1.56	9.37	0.22	NO
Mano Izquierda Abajo - Subir de Frente	39.90	1.66	19.26	0.80	NO
Mano Izquierda Abajo - Mover a Derecha	77.28	1.61	16.98	0.35	NO
Ambas Manos Arriba - Bajar de Frente	37.91	1.35	21.44	0.77	NO

Cuadro 8.68: Evaluación de 16 gestos con perfil Centrado y Rotado mediante Giro eje Y

En la primera gráfica podemos observamos que más del 50 % de los gestos tiene una distancia inferior a 35 metros, y que para más del 50 % de los gestos la distancia entre el gesto detectado y el segundo es menor de 17 metros.

En un principio podría parecer que, en general, se cumple que la distancia entre el gesto detectado y el segundo es mucho menor que la distancia del gesto detectado. Sin embargo, esto no es así, ya que esto sólo es cierto para el 56 % de los gestos. Para el resto de gestos, el 44 % de los gestos, hay una distancia de detección menor que la la distancia entre el gesto detectado y el segundo.

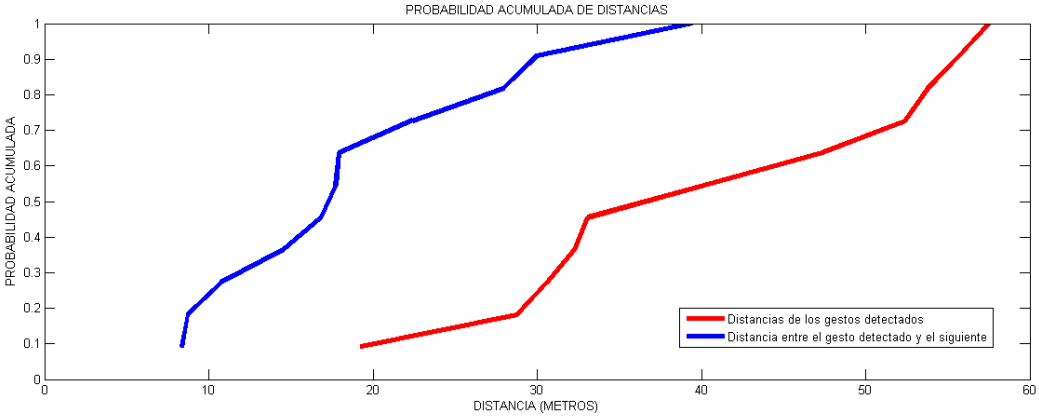


Figura 8.27: Probabilidad acumulada de distancias

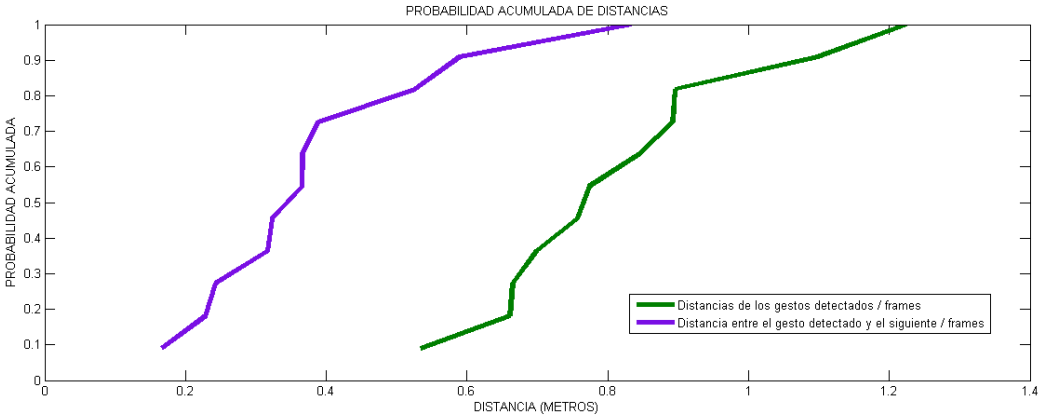


Figura 8.28: Probabilidad acumulada de distancias entre frames

Observando la línea morada en la segunda gráfica, vemos que sólo hay 2 gestos cuya distancias entre el gestos detectado y el segundo más próximo divido entre el número de frames sea menor de 0,2. Y estos dos gestos son gestos detectados correctamente, por lo que los resultados de la matriz de confusión serán peores que los resultados sin umbral.

Sin embargo, observamos que la distancia de los gestos detectados (línea verde) nos aporta una información muy útil. Todos los gestos fallados tienen una distancia de detección superior a 0,9, mientras que los gestos acertados tienen una distancia de detección inferior a 0,49. En este caso el umbral de rechazo $\mathbf{U_{DG}} = 1$ funcionará perfectamente descartando solamente los gestos fallados. Sin embargo, todavía queda un gesto fallado que no ha sido descartado. El umbral de rechazo $\mathbf{U_{DG}}$ ideal en esta evaluación está entre 0,5 y 0,9.

Observando los matrices de confusión, comprobamos que los mejores resultados se obtienen con $\mathbf{U_{DG}} = \mathbf{1}$. En este caso, los resultados son bastante buenos: **acierto** $= \mathbf{94\,\%}$ y valor $\mathbf{F_1} = \mathbf{96\,\%}$. Con un umbral entre 0,5 y 0,9 los resultados serían del 100 %.

$\mathbf{U_{DG}} = \mathbf{1}$

Umbral distancia detección de gestos

	GA	GF	
GDT	(TP) 11	(FP) 1	PRE 0.92
GDS	(FN) 0	(TN) 4	NPV 1
F_1 **0.96**	REC 1	ESP 0.8	ACC **0.94**

$\mathbf{U_{GDS}} = \mathbf{0.2}$

Umbral distancia gesto detectado y segundo

	GA	GF	
GDT	(TP) 9	(FP) 5	PRE 0.64
GDS	(FN) 2	(TN) 0	NPV 0
F_1 **0.72**	REC 0.82	ESP 0	ACC **0.56**

$\mathbf{U_{DG}} = \mathbf{1}$ y $\mathbf{U_{GDS}} = \mathbf{0.2}$

	GA	GF	
GDT	(TP) 9	(FP) 1	PRE 0.9
GDS	(FN) 2	(TN) 4	NPV 0.67
F_1 **0.86**	REC 0.82	ESP 0.8	ACC **0.81**

Sin Umbral

	GA	GF	
GDT	(TP) 11	(FP) 5	PRE 0.69
GDS	(FN) 0	(TN) 0	NPV -
F_1 **0.81**	REC 1	ESP 0	ACC **0.69**

Tabla de Valores F-1 y Acierto para distintos Umbrales

U_{DG}[1]	U_{DS}[2]	F_1[3]	AC[4]
0.02	-	-	0.3125
0.04	-	-	0.3125
0.06	-	-	0.3125
0.08	-	-	0.3125
0.1	-	-	0.3125
0.12	-	-	0.3125
0.14	-	-	0.3125
0.16	-	0.1667	0.3750
0.18	-	0.1667	0.3750
0.2	-	0.1667	0.3750
0.22	-	0.1667	0.3750
0.24	-	0.5333	0.5625
0.26	-	0.6250	0.6250
0.28	-	0.6250	0.6250
0.3	-	0.7059	0.6875
0.32	-	0.8421	0.8125
0.34	-	0.8421	0.8125
0.36	-	0.9524	0.9375
0.38	-	0.9524	0.9375
0.4	-	0.9524	0.9375
0.42	-	0.9524	0.9375
0.44	-	0.9524	0.9375
0.46	-	0.9524	0.9375
0.48	-	0.9524	0.9375
0.5	-	1.0000	1.0000
0.52	-	1.0000	1.0000
0.54	-	1.0000	1.0000
0.56	-	1.0000	1.0000
0.58	-	1.0000	1.0000
0.6	-	1.0000	1.0000
0.62	-	1.0000	1.0000
0.64	-	1.0000	1.0000
0.66	-	1.0000	1.0000
0.68	-	1.0000	1.0000
0.7	-	1.0000	1.0000
0.72	-	1.0000	1.0000
0.74	-	1.0000	1.0000
0.76	-	1.0000	1.0000
0.78	-	1.0000	1.0000
0.8	-	1.0000	1.0000
0.82	-	1.0000	1.0000
0.84	-	1.0000	1.0000
0.86	-	1.0000	1.0000
0.88	-	1.0000	1.0000
0.9	-	1.0000	1.0000
0.92	-	0.9565	0.9375
0.94	-	0.9565	0.9375
0.96	-	0.9565	0.9375
0.98	-	0.9565	0.9375
1	-	0.9565	0.9375

U_{DG}[1]	U_{DS}[2]	F_1[3]	AC[4]
-	0.02	0.8148	0.6875
-	0.04	0.8148	0.6875
-	0.06	0.8148	0.6875
-	0.08	0.8148	0.6875
-	0.1	0.7692	0.6250
-	0.12	0.7692	0.6250
-	0.14	0.7692	0.6250
-	0.16	0.7200	0.5625
-	0.18	0.7200	0.5625
-	0.2	0.7200	0.5625
-	0.22	0.6667	0.5000
-	0.24	0.6957	0.5625
-	0.26	0.6957	0.5625
-	0.28	0.6957	0.5625
-	0.3	0.7273	0.6250
-	0.32	0.6667	0.5625
-	0.34	0.6667	0.5625
-	0.36	0.6316	0.5625
-	0.38	0.5556	0.5000
-	0.4	0.5556	0.5000
-	0.42	0.5556	0.5000
-	0.44	0.5556	0.5000
-	0.46	0.5556	0.5000
-	0.48	0.5556	0.5000
-	0.5	0.5556	0.5000
-	0.52	0.5556	0.5000
-	0.54	0.4706	0.4375
-	0.56	0.4706	0.4375
-	0.58	0.4706	0.4375
-	0.6	0.4706	0.4375
-	0.62	0.3750	0.3750
-	0.64	0.3750	0.3750
-	0.66	0.2667	0.3125
-	0.68	0.2667	0.3125
-	0.7	0.2667	0.3125
-	0.72	0.2667	0.3125
-	0.74	0.2667	0.3125
-	0.76	0.2667	0.3125
-	0.78	0.2857	0.3750
-	0.8	0.2857	0.3750
-	0.82	0.3077	0.4375
-	0.84	0.3077	0.4375
-	0.86	0.3077	0.4375
-	0.88	0.1667	0.3750
-	0.9	-	0.3125
-	0.92	-	0.3125
-	0.94	-	0.3125
-	0.96	-	0.3125
-	0.98	-	0.3125
-	1	-	0.3125

U_{DG}[1]	U_{DS}[2]	F_1[3]	AC[4]
0.05	1	-	0.3125
0.05	0.9	-	0.3125
0.05	0.8	-	0.3125
0.1	0.95	-	0.3125
0.1	0.85	-	0.3125
0.1	0.75	-	0.3125
0.15	0.9	-	0.3125
0.15	0.8	-	0.3125
0.15	0.7	-	0.3125
0.2	0.85	-	0.3125
0.2	0.75	-	0.3125
0.2	0.65	-	0.3125
0.25	0.8	0.1667	0.3750
0.25	0.7	0.1667	0.3750
0.25	0.6	0.4286	0.5000
0.3	0.75	0.3077	0.4375
0.3	0.65	0.3077	0.4375
0.3	0.55	0.5333	0.5625
0.35	0.7	0.3077	0.4375
0.35	0.6	0.5333	0.5625
0.35	0.5	0.6250	0.6250
0.4	0.65	0.3077	0.4375
0.4	0.55	0.5333	0.5625
0.4	0.45	0.6250	0.6250
0.45	0.6	0.5333	0.5625
0.45	0.5	0.6250	0.6250
0.45	0.4	0.6250	0.6250
0.5	0.55	0.5333	0.5625
0.5	0.45	0.6250	0.6250
0.5	0.35	0.7778	0.7500
0.55	0.5	0.6250	0.6250
0.55	0.4	0.6250	0.6250
0.55	0.3	0.8421	0.8125
0.6	0.45	0.6250	0.6250
0.6	0.35	0.7778	0.7500
0.6	0.25	0.8421	0.8125
0.65	0.4	0.6250	0.6250
0.65	0.3	0.8421	0.8125
0.65	0.2	0.9000	0.8750
0.7	0.35	0.7778	0.7500
0.7	0.25	0.8421	0.8125
0.7	0.15	0.9000	0.8750
0.75	0.3	0.8421	0.8125
0.75	0.2	0.9000	0.8750
0.75	0.1	0.9524	0.9375
0.8	0.25	0.8421	0.8125
0.8	0.15	0.9000	0.8750
0.8	0.05	1.0000	1.0000
0.85	0.2	0.9000	0.8750
0.85	0.1	0.9524	0.9375

1 - $\mathbf{U_{DG}}$ - Umbral de distancia del gesto detectado
2 - $\mathbf{U_{GDS}}$ - Umbral de distancia entre el gesto detectado y el segundo
3 - $\mathbf{F_1}$ - Valor F-1
4 - **AC** - Acierto

8.3.4. Esqueleto Centrado y Rotado - Variaciones DTW

En las evaluaciones se han analizado dos tipos de variaciones en el algoritmo DTW:

- Desplazamiento (Sakoe-Chiba) = 2
- Desplazamiento (Itakura) = 2

Estas variaciones se han aplicado con el preprocesamiento de Centrado y Rotado. En ambas las conclusiones han sido las mismas: no se aprecia ningún cambio en los resultados al usar estas variaciones. Hay que tener en cuenta que las secuencias con las que trabajamos, tienen 30 frames por segundos, y considerando que un gesto dure 2 gestos, hay 60 esqueletos que tienen que ser alineados. Con tan pocos elementos para ser alineados, y teniendo además en cuenta que entre un frame y otro siempre hay movimiento (sino no se detectaría como un gesto), las restricciones que imponen las variaciones DTW no afectan prácticamente en nada.

Esqueleto Centrado y Rotado - Desplazamiento (Sakoe-Chiba) = 2

Esta variante del algoritmo DTW impone que el máximo deplazamiento lateral al alinear las secuencias de esqueletos sea de 2 esqueletos. Es decir, que como mucho 2 esqueletos del gesto realizado de la evaluación se van a poder alinear con el mismo esqueleto del desto realizado en el entrenamiento.

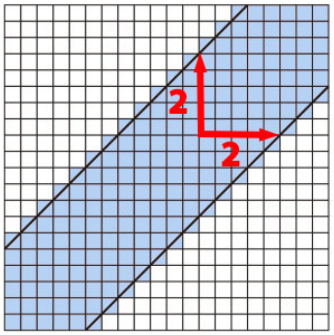


Figura 8.29: Desplazamiento = 2

Como vemos en la siguiente gráfica de distancias de los gestos detectados para la evaluación de 33 gestos, las distancias de ambos perfiles son prácticamente

idénticas.

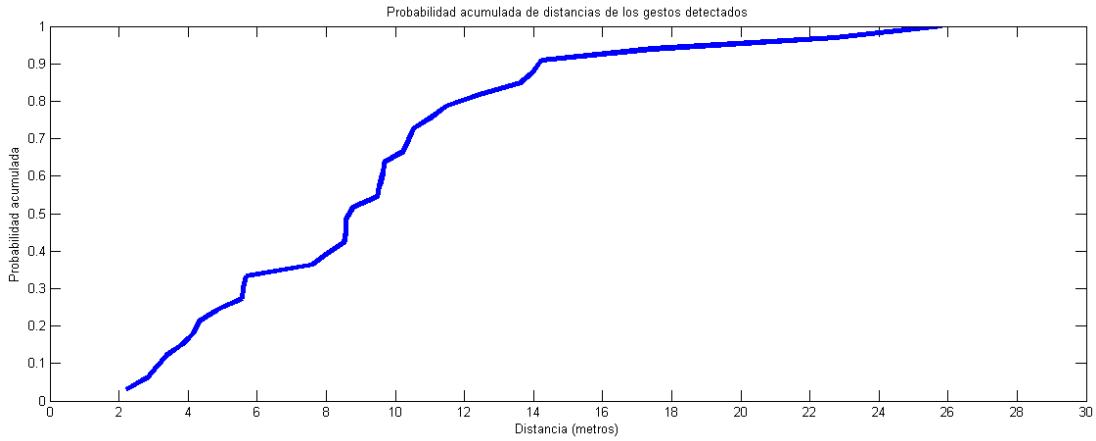


Figura 8.30: Sin variación DTW

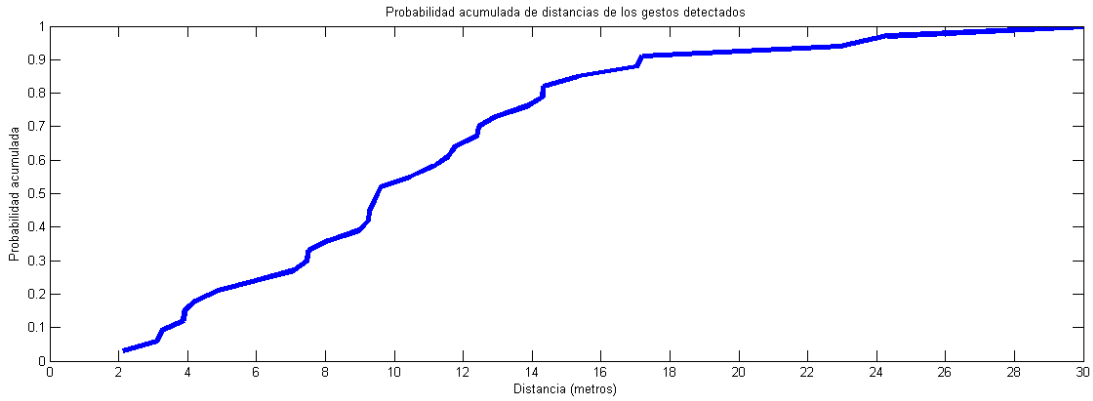


Figura 8.31: Desplazamiento = 2

La fiabilidad es la misma que el perfil de *Centrado y Rotado*, y las distancias son prácticamente idénticas, por lo que no encontramos ninguna ventaja en utilizar esta variación del DTW.

Esqueleto Centrado y Rotado - Desplazamiento (Itakura) = 2

Como vemos en la siguiente gráfica de distancias de los gestos detectados para la evaluación de 33 gestos, las distancias de ambos perfiles son prácticamente idénticas.

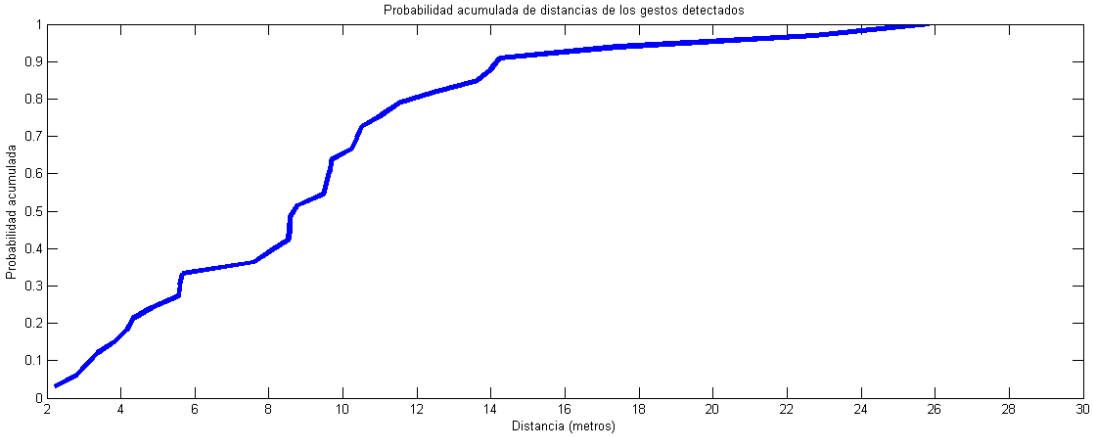


Figura 8.32: Sin variación DTW

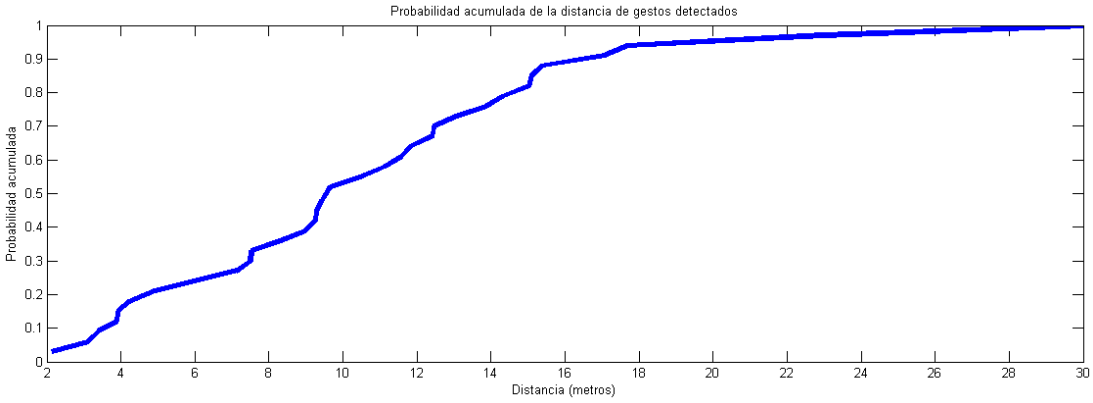


Figura 8.33: Desplazamiento = 2

La fiabilidad es la misma que el perfil de *Centrado y Rotado*, y las distancias son prácticamente idénticas, por lo que no encontramos ninguna ventaja en utilizar esta variación del DTW.

8.3.5. Evaluaciones con el esqueleto centrado y rotado - Distancia selectiva con umbral

El perfil de *centrado y rotado - Distancia selectiva con umbral*, el preprocesamiento es el mismo que en el perfil de *centrado y rotado mediante cambio de base*, pero ahora el tipo de distancia es Distancia Selectiva con Umbral (Pág. 60), que sólo tiene en cuenta las distancias de las componentes del cuerpo que superen un umbral.

$$d_u(a,b) = \sum_i d(p_{ia}, p_{ib}) \quad \forall\ i \in s\ /\ d(p_{ia}, p_{ib}) > u_d$$

$$\mathbf{s} = \text{hombros, codos, muñecas y manos}$$

$$\mathbf{u_d} = \text{umbral de distancia}$$

En esta evaluación se ha utilizado un **umbral de distancia** igual a **0.3**.

Evaluación de 11 gestos

GESTO	DISTANCIA GESTO DETECTADO	DISTANCIA GESTO DETECTADO / FRAMES	DISTANCIA ENTRE GESTO DETECTADO Y SEGUNDO	DISTANCIA ENTRE GESTO DETECTADO Y SEGUNDO / FRAMES	ACIERTO
Ambas Manos Abajo - Subir y Bajar	0,00	0,00	29,80	0,66	SI
Ambas Manos Abajo - Subir y Bajar de Frente	0,00	0,00	37,68	0,82	SI
Ambas Manos Abajo - Subir por completo y Bajar de Frente	0,00	0,00	35,75	0,58	SI
Mano Derecha Abajo - Subir y Bajar	0,00	0,00	31,40	0,67	SI
Mano Derecha Abajo - Subir y Bajar de Frente	0,00	0,00	11,72	0,23	SI
Mano Derecha Abajo - Subir por completo y Bajar de Frente	0,00	0,00	13,65	0,22	SI
Mano Izquierda Abajo - Subir y Bajar	0,00	0,00	27,12	0,66	SI
Mano Izquierda Abajo - Subir y Bajar de Frente	0,00	0,00	14,11	0,35	SI
Mano Izquierda Abajo - Subir por completo y Bajar de Frente	0,00	0,00	12,71	0,23	SI
Saludo Mano Derecha	0,00	0,00	35,15	0,50	SI
Saludo Mano Izquierda	0,00	0,00	37,66	0,52	SI

Cuadro 8.74: Evaluación de 11 gestos - Centrado y Rotado con distancias al cubo

Como vemos en la tabla, este perfil, al igual que el perfil de *centrado y rotado mediante cambio de base*, obtiene un acierto del 100 %. Aunque era de esperar que las distancias de los gestos detectados fueran ahora menores, vemos que todas las distancias de los gestos detectados son **cero**. Las distancias entre el gesto detectado y el segundo sin embargo se mantienen como en otros perfiles, por encima de **0.2**.

Observando la gráficas de Probabilidad acumulada de distancias de gesto detectados, comprobamos que efectivamente todas las distancias de los gestos detectados son nulas. Sin embargo, las distancias entre el gesto detectado y el segundo están por encima de los 10 metros (línea azul) o 0,2 para distancias normalizadas por el número de frames (línea morada).

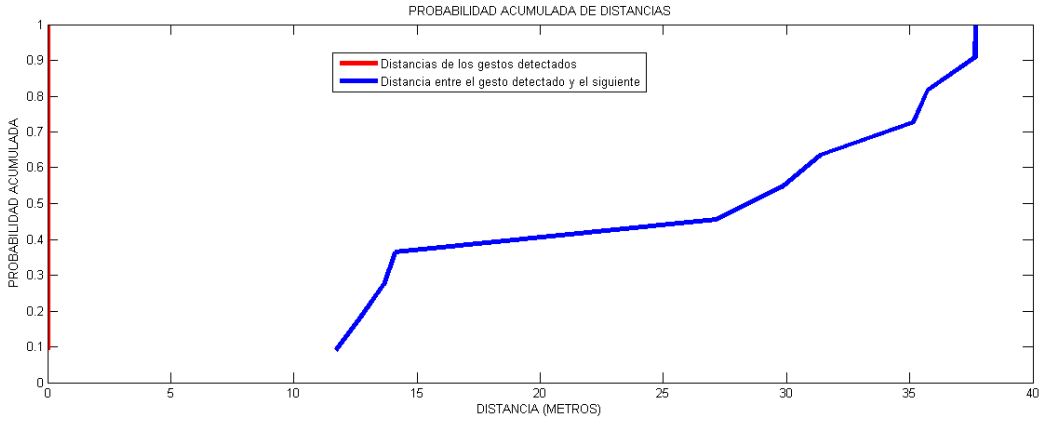


Figura 8.34: Probabilidad acumulada de distancias

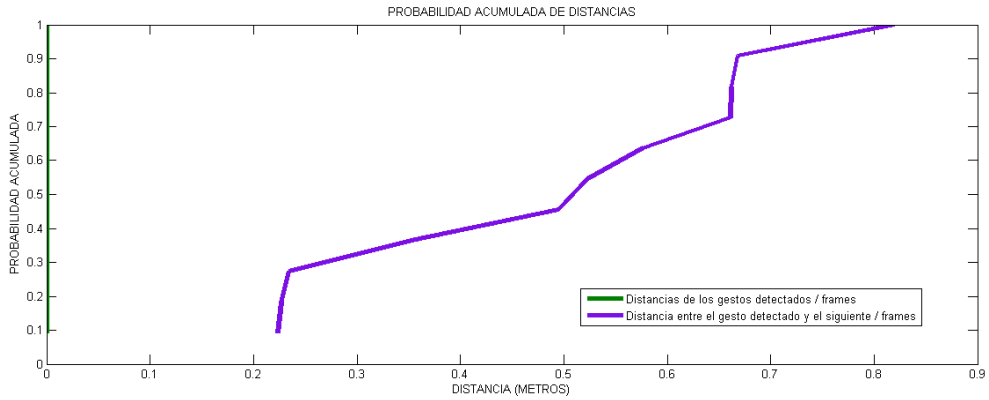


Figura 8.35: Probabilidad acumulada de distancias entre frames

En esta evaluación todos los gestos han sido detectados correctamente, con unas distancias de detección nulas. Para todos los casos los resultados son muy buenos: **acierto** $= \mathbf{100\,\%}$ y valor $\mathbf{F_1} = \mathbf{100\,\%}$.

$U_{DG} = 0.1$

Umbral distancia deteccion de gestos

	GA	GF	
GDT	(TP) 11	(FP) 0	PRE 1
GDS	(FN) 0	(TN) 0	NPV -
F_1 1	REC 1	ESP -	ACC 1

$U_{GDS} = 0.2$

Umbral distancia gesto detectado y segundo

	GA	GF	
GDT	(TP) 11	(FP) 0	PRE 1
GDS	(FN) 0	(TN) 0	NPV -
F_1 1	REC 1	ESP -	ACC 1

$U_{DG} = 0.1$ y $U_{GDS} = 0.2$

	GA	GF	
GDT	(TP) 11	(FP) 0	PRE 1
GDS	(FN) 0	(TN) 0	NPV -
F_1 1	REC 1	ESP -	ACC 1

Sin Umbral

	GA	GF	
GDT	(TP) 11	(FP) 0	PRE 1
GDS	(FN) 0	(TN) 0	NPV -
F_1 1	REC 1	ESP -	ACC 1

Tabla de Valores F-1 y Acierto para distintos Umbrales

U_{DG} [1]	U_{DS} [2]	F_1 [3]	AC [4]
0.02	-	1.0000	1.0000
0.04	-	1.0000	1.0000
0.06	-	1.0000	1.0000
0.08	-	1.0000	1.0000
0.1	-	1.0000	1.0000
0.12	-	1.0000	1.0000
0.14	-	1.0000	1.0000
0.16	-	1.0000	1.0000
0.18	-	1.0000	1.0000
0.2	-	1.0000	1.0000
0.22	-	1.0000	1.0000
0.24	-	1.0000	1.0000
0.26	-	1.0000	1.0000
0.28	-	1.0000	1.0000
0.3	-	1.0000	1.0000
0.32	-	1.0000	1.0000
0.34	-	1.0000	1.0000
0.36	-	1.0000	1.0000
0.38	-	1.0000	1.0000
0.4	-	1.0000	1.0000
0.42	-	1.0000	1.0000
0.44	-	1.0000	1.0000
0.46	-	1.0000	1.0000
0.48	-	1.0000	1.0000
0.5	-	1.0000	1.0000
0.52	-	1.0000	1.0000
0.54	-	1.0000	1.0000
0.56	-	1.0000	1.0000
0.58	-	1.0000	1.0000
0.6	-	1.0000	1.0000
0.62	-	1.0000	1.0000
0.64	-	1.0000	1.0000
0.66	-	1.0000	1.0000
0.68	-	1.0000	1.0000
0.7	-	1.0000	1.0000
0.72	-	1.0000	1.0000
0.74	-	1.0000	1.0000
0.76	-	1.0000	1.0000
0.78	-	1.0000	1.0000
0.8	-	1.0000	1.0000
0.82	-	1.0000	1.0000
0.84	-	1.0000	1.0000
0.86	-	1.0000	1.0000
0.88	-	1.0000	1.0000
0.9	-	1.0000	1.0000
0.92	-	1.0000	1.0000
0.94	-	1.0000	1.0000
0.96	-	1.0000	1.0000
0.98	-	1.0000	1.0000
1	-	1.0000	1.0000

U_{DG} [1]	U_{DS} [2]	F_1 [3]	AC [4]
-	0.02	1.0000	1.0000
-	0.04	1.0000	1.0000
-	0.06	1.0000	1.0000
-	0.08	1.0000	1.0000
-	0.1	1.0000	1.0000
-	0.12	1.0000	1.0000
-	0.14	1.0000	1.0000
-	0.16	1.0000	1.0000
-	0.18	1.0000	1.0000
-	0.2	1.0000	1.0000
-	0.22	1.0000	1.0000
-	0.24	0.8421	0.7273
-	0.26	0.8421	0.7273
-	0.28	0.8421	0.7273
-	0.3	0.8421	0.7273
-	0.32	0.8421	0.7273
-	0.34	0.8421	0.7273
-	0.36	0.7778	0.6364
-	0.38	0.7778	0.6364
-	0.4	0.7778	0.6364
-	0.42	0.7778	0.6364
-	0.44	0.7778	0.6364
-	0.46	0.7778	0.6364
-	0.48	0.7778	0.6364
-	0.5	0.7059	0.5455
-	0.52	0.7059	0.5455
-	0.54	0.6250	0.4545
-	0.56	0.6250	0.4545
-	0.58	0.5333	0.3636
-	0.6	0.5333	0.3636
-	0.62	0.5333	0.3636
-	0.64	0.5333	0.3636
-	0.66	0.5333	0.3636
-	0.68	0.1667	0.0909
-	0.7	0.1667	0.0909
-	0.72	0.1667	0.0909
-	0.74	0.1667	0.0909
-	0.76	0.1667	0.0909
-	0.78	0.1667	0.0909
-	0.8	0.1667	0.0909
-	0.82	-	0.0000
-	0.84	-	0.0000
-	0.86	-	0.0000
-	0.88	-	0.0000
-	0.9	-	0.0000
-	0.92	-	0.0000
-	0.94	-	0.0000
-	0.96	-	0.0000
-	0.98	-	0.0000
-	1	-	0.0000

U_{DG} [1]	U_{DS} [2]	F_1 [3]	AC [4]
0.05	1	-	0.0000
0.05	0.9	-	0.0000
0.05	0.8	0.1667	0.0909
0.1	0.95	-	0.0000
0.1	0.85	-	0.0000
0.1	0.75	0.1667	0.0909
0.15	0.9	-	0.0000
0.15	0.8	0.1667	0.0909
0.15	0.7	0.1667	0.0909
0.2	0.85	-	0.0000
0.2	0.75	0.1667	0.0909
0.2	0.65	0.5333	0.3636
0.25	0.8	0.1667	0.0909
0.25	0.7	0.1667	0.0909
0.25	0.6	0.5333	0.3636
0.3	0.75	0.1667	0.0909
0.3	0.65	0.5333	0.3636
0.3	0.55	0.6250	0.4545
0.35	0.7	0.1667	0.0909
0.35	0.6	0.5333	0.3636
0.35	0.5	0.7059	0.5455
0.4	0.65	0.5333	0.3636
0.4	0.55	0.6250	0.4545
0.4	0.45	0.7778	0.6364
0.45	0.6	0.5333	0.3636
0.45	0.5	0.7059	0.5455
0.45	0.4	0.7778	0.6364
0.5	0.55	0.6250	0.4545
0.5	0.45	0.7778	0.6364
0.5	0.35	0.8421	0.7273
0.55	0.5	0.7059	0.5455
0.55	0.4	0.7778	0.6364
0.55	0.3	0.8421	0.7273
0.6	0.45	0.7778	0.6364
0.6	0.35	0.8421	0.7273
0.6	0.25	0.8421	0.7273
0.65	0.4	0.7778	0.6364
0.65	0.3	0.8421	0.7273
0.65	0.2	1.0000	1.0000
0.7	0.35	0.8421	0.7273
0.7	0.25	0.8421	0.7273
0.7	0.15	1.0000	1.0000
0.75	0.3	0.8421	0.7273
0.75	0.2	1.0000	1.0000
0.75	0.1	1.0000	1.0000
0.8	0.25	0.8421	0.7273
0.8	0.15	1.0000	1.0000
0.8	0.05	1.0000	1.0000
0.85	0.2	1.0000	1.0000
0.85	0.1	1.0000	1.0000

1 - $\mathbf{U_{DG}}$ - Umbral de distancia del gesto detectado
2 - $\mathbf{U_{GDS}}$ - Umbral de distancia entre el gesto detectado y el segundo
3 - $\mathbf{F_1}$ - Valor F-1
4 - **AC** - Acierto

Evaluación de 11 gestos con el cuerpo girado 45º

Esta evaluación, al igual que con el perfil de *Centrado y Rotado mediante cambio de base*, tiene una porcentaje de acierto alto, del 82 %, Además, las distancias de detección de gestos son todas muy bajas (todas cero excepto 4, con distancias inferiores a 0,12).

GESTO	DISTANCIA GESTO DETECTADO	DISTANCIA GESTO DETECTADO / FRAMES	DISTANCIA ENTRE GESTO DETECTADO Y SEGUNDO	DISTANCIA ENTRE GESTO DETECTADO Y SEGUNDO / FRAMES	ACIERTO
Ambas Manos Abajo - Subir y Bajar	0,00	0,00	35,03	0,47	SI
Ambas Manos Abajo - Subir y Bajar de Frente	5,74	0,12	5,22	0,11	NO
Ambas Manos Abajo - Subir por completo y Bajar de Frente	2,70	0,06	13,66	0,28	SI
Mano Derecha Abajo - Subir y Bajar	0,00	0,00	27,86	0,77	SI
Mano Derecha Abajo - Subir y Bajar de Frente	2,23	0,06	0,21	0,01	SI
Mano Derecha Abajo - Subir por completo y Bajar de Frente	0,60	0,01	10,01	0,22	SI
Mano Izquierda Abajo - Subir y Bajar	0,00	0,00	12,27	0,34	SI
Mano Izquierda Abajo - Subir y Bajar de Frente	0,00	0,00	10,33	0,22	SI
Mano Izquierda Abajo - Subir por completo y Bajar de Frente	0,00	0,00	3,73	0,07	NO
Saludo Mano Derecha	0,00	0,00	27,31	0,38	SI
Saludo Mano Izquierda	0,00	0,00	21,67	0,39	SI

Cuadro 8.80: Evaluación de 11 gestos con el cuerpo girado 45º con perfil Centrado y Rotado con distancias al cubo

En esta evaluación se han fallado dos gestos. Uno de ellos es el mismo que se fallaba con el perfil de *Centrado y Rotado mediante Cambio de Base*, y el otro, aunque se acertaba, tenía una distancia entre el gesto detectado y el segundo más próximo de sólo 0,05.

La Probabilidad acumulada de distancias de gesto detectados y la Probabilidad acumulada de distancias de gesto detectado y segundo nos dan una idea del porcentaje de gestos que están por debajo de una determinada distancia:

En general observamos que las distancias de los gestos detectados son valores muy pequeños. En la primera gráfica, observando la línea roja, podemos ver que más del 65 % de los gestos tienen una distancia nula.

Las distancias entre cada gesto detectado y el siguiente, el segundo gesto más parecido, tiene también valores pequeños, aunque mucho más grandes que las distancias de los gestos detectados. Observando la línea azul, podemos ver que para más del 50 % de los gestos la distancia entre el gesto detectado y el segundo es menor de 12 metros.

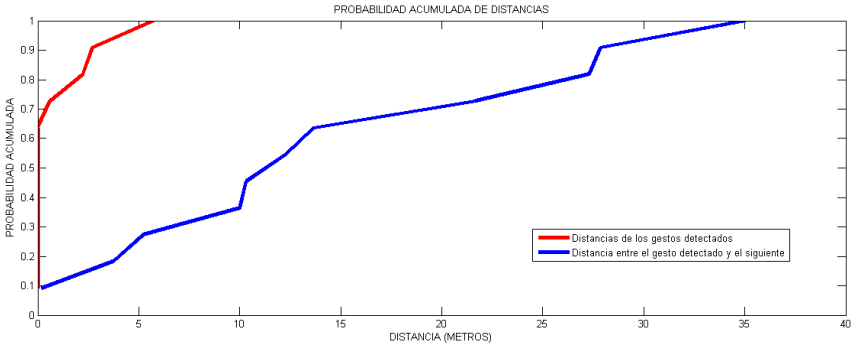


Figura 8.36: Probabilidad acumulada de distancias

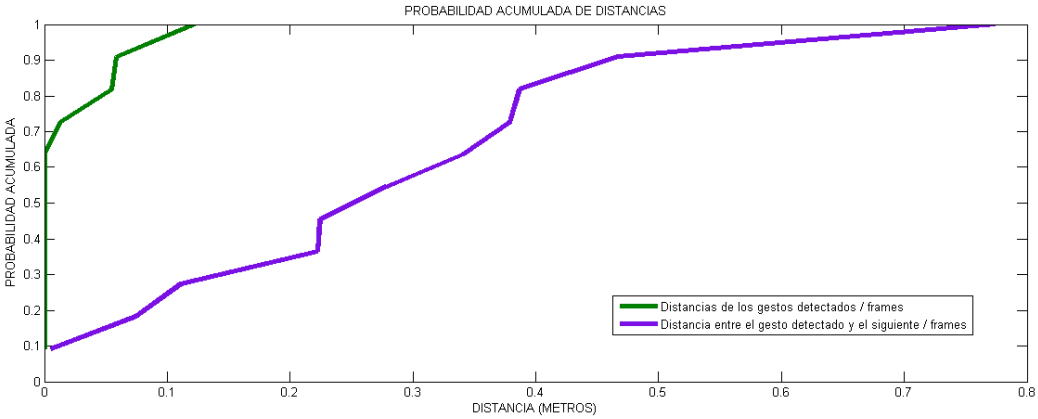


Figura 8.37: Probabilidad acumulada de distancias entre frames

Como vemos en la línea verde de la segunda gráfica, todas las distancias están por debajo de 0,1, excepto un gesto fallado que está por encima de 0,1. Por tanto, los resultados de la matriz de confusión para $\mathbf{U_{DG}} = \mathbf{0.1}$ serán mejores que los resultados sin umbral.

Observando la línea morada, vemos que hay 3 gestos por debajo del 0,2: 1 gestos acertados y 2 fallados. Los resultados de la matriz de confusión para $\mathbf{U_{GDS}} = \mathbf{0.2}$ serán mejores que los resultados sin umbral.

Observando los matrices de confusión, comprobamos que los mejores resultados se obtienen usando umbral de rechazo. Tanto para $\mathbf{U_{DG}} = \mathbf{0.1}$ como para $\mathbf{U_{GDS}} = \mathbf{0.2}$ los resultados son bastante buenos: **acierto** $= \mathbf{91\,\%}$ y valor $\mathbf{F_1} = \mathbf{95\,\%}$.

$\mathbf{U_{DG}} = \mathbf{0.1}$

Umbral distancia deteccion de gestos

	GA	GF	
GDT	(TP) 9	(FP) 1	PRE 0.9
GDS	(FN) 0	(TN) 1	NPV 1
F_1 0.95	REC 1	ESP 0.5	ACC 0.91

$\mathbf{U_{GDS}} = \mathbf{0.2}$

Umbral distancia gesto detectado y segundo

	GA	GF	
GDT	(TP) 8	(FP) 0	PRE 1
GDS	(FN) 1	(TN) 2	NPV 0.67
F_1 0.94	REC 0.89	ESP 1	ACC 0.91

$\mathbf{U_{DG}} = \mathbf{0.1}$ y $\mathbf{U_{GDS}} = \mathbf{0.2}$

	GA	GF	
GDT	(TP) 8	(FP) 0	PRE 1
GDS	(FN) 1	(TN) 2	NPV 0.67
F_1 0.94	REC 0.89	ESP 1	ACC 0.91

Sin Umbral

	GA	GF	
GDT	(TP) 9	(FP) 2	PRE 0.82
GDS	(FN) 0	(TN) 0	NPV -
F_1 0.9	REC 1	ESP 0	ACC 0.82

Tabla de Valores F-1 y Acierto para distintos Umbrales

U_{DG} [1]	U_{DS} [2]	F_1 [3]	AC [4]
0.02	-	0.8235	0.7273
0.04	-	0.8235	0.7273
0.06	-	0.9474	0.9091
0.08	-	0.9474	0.9091
0.1	-	0.9474	0.9091
0.12	-	0.9474	0.9091
0.14	-	0.9000	0.8182
0.16	-	0.9000	0.8182
0.18	-	0.9000	0.8182
0.2	-	0.9000	0.8182
0.22	-	0.9000	0.8182
0.24	-	0.9000	0.8182
0.26	-	0.9000	0.8182
0.28	-	0.9000	0.8182
0.3	-	0.9000	0.8182
0.32	-	0.9000	0.8182
0.34	-	0.9000	0.8182
0.36	-	0.9000	0.8182
0.38	-	0.9000	0.8182
0.4	-	0.9000	0.8182
0.42	-	0.9000	0.8182
0.44	-	0.9000	0.8182
0.46	-	0.9000	0.8182
0.48	-	0.9000	0.8182
0.5	-	0.9000	0.8182
0.52	-	0.9000	0.8182
0.54	-	0.9000	0.8182
0.56	-	0.9000	0.8182
0.58	-	0.9000	0.8182
0.6	-	0.9000	0.8182
0.62	-	0.9000	0.8182
0.64	-	0.9000	0.8182
0.66	-	0.9000	0.8182
0.68	-	0.9000	0.8182
0.7	-	0.9000	0.8182
0.72	-	0.9000	0.8182
0.74	-	0.9000	0.8182
0.76	-	0.9000	0.8182
0.78	-	0.9000	0.8182
0.8	-	0.9000	0.8182
0.82	-	0.9000	0.8182
0.84	-	0.9000	0.8182
0.86	-	0.9000	0.8182
0.88	-	0.9000	0.8182
0.9	-	0.9000	0.8182
0.92	-	0.9000	0.8182
0.94	-	0.9000	0.8182
0.96	-	0.9000	0.8182
0.98	-	0.9000	0.8182
1	-	0.9000	0.8182

U_{DG} [1]	U_{DS} [2]	F_1 [3]	AC [4]
-	0.02	0.8421	0.7273
-	0.04	0.8421	0.7273
-	0.06	0.8421	0.7273
-	0.08	0.8889	0.8182
-	0.1	0.8889	0.8182
-	0.12	0.9412	0.9091
-	0.14	0.9412	0.9091
-	0.16	0.9412	0.9091
-	0.18	0.9412	0.9091
-	0.2	0.9412	0.9091
-	0.22	0.9412	0.9091
-	0.24	0.8000	0.7273
-	0.26	0.8000	0.7273
-	0.28	0.7143	0.6364
-	0.3	0.7143	0.6364
-	0.32	0.7143	0.6364
-	0.34	0.7143	0.6364
-	0.36	0.6154	0.5455
-	0.38	0.5000	0.4545
-	0.4	0.3636	0.3636
-	0.42	0.3636	0.3636
-	0.44	0.3636	0.3636
-	0.46	0.3636	0.3636
-	0.48	0.2000	0.2727
-	0.5	0.2000	0.2727
-	0.52	0.2000	0.2727
-	0.54	0.2000	0.2727
-	0.56	0.2000	0.2727
-	0.58	0.2000	0.2727
-	0.6	0.2000	0.2727
-	0.62	0.2000	0.2727
-	0.64	0.2000	0.2727
-	0.66	0.2000	0.2727
-	0.68	0.2000	0.2727
-	0.7	0.2000	0.2727
-	0.72	0.2000	0.2727
-	0.74	0.2000	0.2727
-	0.76	0.2000	0.2727
-	0.78	-	0.1818
-	0.8	-	0.1818
-	0.82	-	0.1818
-	0.84	-	0.1818
-	0.86	-	0.1818
-	0.88	-	0.1818
-	0.9	-	0.1818
-	0.92	-	0.1818
-	0.94	-	0.1818
-	0.96	-	0.1818
-	0.98	-	0.1818
-	1	-	0.1818

U_{DG} [1]	U_{DS} [2]	F_1 [3]	AC [4]
0.05	1	-	0.1818
0.05	0.9	-	0.1818
0.05	0.8	-	0.1818
0.1	0.95	-	0.1818
0.1	0.85	-	0.1818
0.1	0.75	0.2000	0.2727
0.15	0.9	-	0.1818
0.15	0.8	-	0.1818
0.15	0.7	0.2000	0.2727
0.2	0.85	-	0.1818
0.2	0.75	0.2000	0.2727
0.2	0.65	0.2000	0.2727
0.25	0.8	-	0.1818
0.25	0.7	0.2000	0.2727
0.25	0.6	0.2000	0.2727
0.3	0.75	0.2000	0.2727
0.3	0.65	0.2000	0.2727
0.3	0.55	0.2000	0.2727
0.35	0.7	0.2000	0.2727
0.35	0.6	0.2000	0.2727
0.35	0.5	0.2000	0.2727
0.4	0.65	0.2000	0.2727
0.4	0.55	0.2000	0.2727
0.4	0.45	0.3636	0.3636
0.45	0.6	0.2000	0.2727
0.45	0.5	0.2000	0.2727
0.45	0.4	0.3636	0.3636
0.5	0.55	0.2000	0.2727
0.5	0.45	0.3636	0.3636
0.5	0.35	0.6154	0.5455
0.55	0.5	0.2000	0.2727
0.55	0.4	0.3636	0.3636
0.55	0.3	0.7143	0.6364
0.6	0.45	0.3636	0.3636
0.6	0.35	0.6154	0.5455
0.6	0.25	0.8000	0.7273
0.65	0.4	0.3636	0.3636
0.65	0.3	0.7143	0.6364
0.65	0.2	0.9412	0.9091
0.7	0.35	0.6154	0.5455
0.7	0.25	0.8000	0.7273
0.7	0.15	0.9412	0.9091
0.75	0.3	0.7143	0.6364
0.75	0.2	0.9412	0.9091
0.75	0.1	0.8889	0.8182
0.8	0.25	0.8000	0.7273
0.8	0.15	0.9412	0.9091
0.8	0.05	0.8421	0.7273
0.85	0.2	0.9412	0.9091
0.85	0.1	0.8889	0.8182

1 - $\mathbf{U_{DG}}$ - Umbral de distancia del gesto detectado
2 - $\mathbf{U_{GDS}}$ - Umbral de distancia entre el gesto detectado y el segundo
3 - $\mathbf{F_1}$ - Valor F-1
4 - **AC** - Acierto

Evaluación de 16 gestos con 5 gestos inexistentes

El porcentaje de acierto es del 69 % (11 gestos acertados de 16) debido a que 5 gestos son inexistentes y se van a fallar seguro. Sin embargo, es interesante analizar el umbral de rechazo para ver en qué casos esos gestos van a ser descartados por no superar el umbral de rechazo, y pasar de Falsos Positivos (FP) a Verdaderos Negativos (TN).

GESTO	DISTANCIA GESTO DETECTADO	DISTANCIA GESTO DETECTADO / FRAMES	DISTANCIA ENTRE GESTO DETECTADO Y SEGUNDO	DISTANCIA ENTRE GESTO DETECTADO Y SEGUNDO / FRAMES	ACIERTO
Ambas Manos Abajo - Subir y Bajar	0,00	0,00	35,32	0,77	SI
Ambas Manos Abajo - Subir y Bajar de Frente	0,00	0,00	16,92	0,38	SI
Ambas Manos Abajo - Subir por completo y Bajar de Frente	4,90	0,09	3,01	0,06	SI
Mano Derecha Abajo - Subir y Bajar	0,00	0,00	28,39	0,71	SI
Mano Derecha Abajo - Subir y Bajar de Frente	0,00	0,00	8,41	0,18	SI
Mano Derecha Abajo - Subir por completo y Bajar de Frente	0,00	0,00	2,59	0,05	SI
Mano Izquierda Abajo - Subir y Bajar	0,00	0,00	13,05	0,32	SI
Mano Izquierda Abajo - Subir y Bajar de Frente	0,00	0,00	3,32	0,08	SI
Mano Izquierda Abajo - Subir por completo y Bajar de Frente	0,00	0,00	5,45	0,11	SI
Saludo Mano Derecha	4,04	0,06	19,23	0,30	SI
Saludo Mano Izquierda	0,00	0,00	17,03	0,28	SI
Mano Derecha Abajo - Subir y Bajar Cruzado	17,54	0,39	5,96	0,13	NO
Mano Derecha Abajo - Mover a Izquierda y a Derecha	31,17	0,74	14,44	0,34	NO
Mano Izquierda Abajo - Subir de Frente	27,84	1,16	10,84	0,45	NO
Mano Izquierda Abajo - Mover a Derecha	55,77	1,16	9,57	0,20	NO
Ambas Manos Arriba - Bajar de Frente	12,97	0,46	19,49	0,70	NO

Cuadro 8.86: Evaluación de 16 gestos con perfil Centrado y Rotado con distancias al cubo

En la primera gráfica, observando la línea roja, podemos ver que para un 50 % de los gestos la distancia de los gestos detectados es nula. Además, todos los gestos detectados que superan los 10 metros de distancia son gestos fallados.

Sin embargo, con la distancia entre el gesto detectado y el segundo más próximo no hay un patrón similar. Las distancias de los gestos acertados y fallados están mezcladas. No se puede decir que en general las distancias entre el gesto detectado y el segundo más próximo de los gestos fallados sean mayores o menores que la de los gestos acertados.

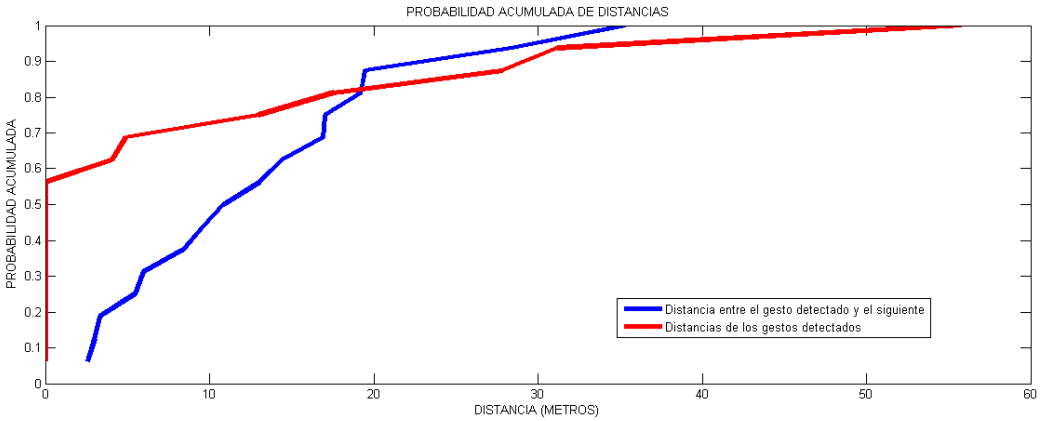


Figura 8.38: Probabilidad acumulada de distancias

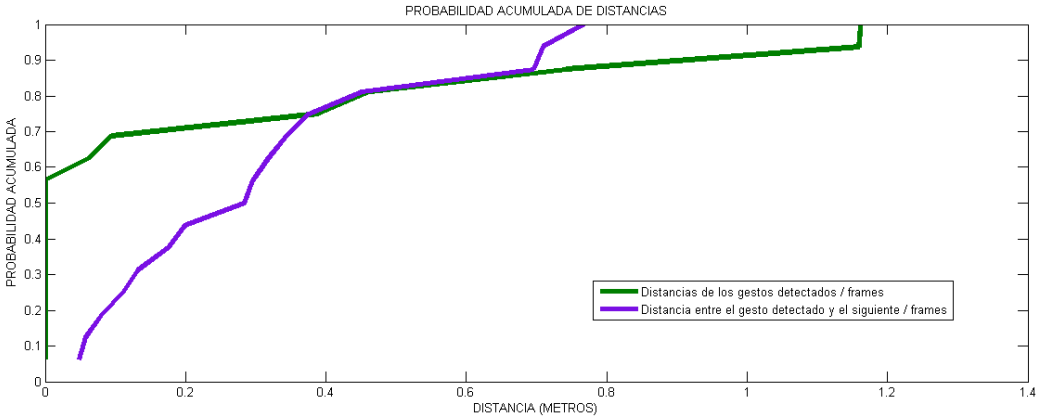


Figura 8.39: Probabilidad acumulada de distancias entre frames

Observando la línea morada en la segunda gráfica, vemos que hay 7 gestos cuya distancia entre el gestos detectado y el segundo más próximo divido entre el número de frames sea menor de 0,2. Sin embargo, de estos 7 gestos, 5 han sido detectados correctamente y 2 han sido detectados incorrectamenet. Por tanto, el umbral $\mathbf{U_{GDS}} = \mathbf{0.2}$ generará unos resultados peores que sin umbral.

Sin embargo, observamos que la distancia de los gestos detectados (línea verde) nos aporta una información muy útil. Todos los gestos fallados tienen una distancia de detección superior a **0.38**, mientras que los gestos acertados tienen una distancia de detección inferior a **0.09**. En este caso el umbral de rechazo $\mathbf{U_{DG}} = 0{,}1$ funcionará perfectamente descartando todos los gestos fallados.

Observando los matrices de confusión, comprobamos que los mejores resultados se obtienen con $\mathbf{U_{DG}} = \mathbf{0.1}$. En este caso, los resultados son inmejorables: **acierto** $= \mathbf{100\,\%}$ y valor $\mathbf{F_1} = \mathbf{100\,\%}$.

$\mathbf{U_{DG}} = \mathbf{0.1}$

Umbral distancia deteccion de gestos

	GA	GF	
GDT	(TP) 11	(FP) 0	PRE 1
GDS	(FN) 0	(TN) 5	NPV 1
F_1 1	REC 1	ESP 1	ACC 1

$\mathbf{U_{GDS}} = \mathbf{0.2}$

Umbral distancia gesto detectado y segundo

	GA	GF	
GDT	(TP) 6	(FP) 3	PRE 0.67
GDS	(FN) 5	(TN) 2	NPV 0.29
F_1 0.6	REC 0.55	ESP 0.4	ACC 0.5

$\mathbf{U_{DG}} = \mathbf{0.1}$ y $\mathbf{U_{GDS}} = \mathbf{0.2}$

	GA	GF	
GDT	(TP) 6	(FP) 0	PRE 1
GDS	(FN) 5	(TN) 5	NPV 0.5
F_1 0.71	REC 0.55	ESP 1	ACC 0.69

Sin Umbral

	GA	GF	
GDT	(TP) 11	(FP) 5	PRE 0.69
GDS	(FN) 0	(TN) 0	NPV -
F_1 0.81	REC 1	ESP 0	ACC 0.69

Tabla de Valores F-1 y Acierto para distintos Umbrales

U_{DG} [1]	U_{DS} [2]	F_1 [3]	AC [4]
0.02	-	0.9000	0.8750
0.04	-	0.9000	0.8750
0.06	-	0.9000	0.8750
0.08	-	0.9524	0.9375
0.1	-	1.0000	1.0000
0.12	-	1.0000	1.0000
0.14	-	1.0000	1.0000
0.16	-	1.0000	1.0000
0.18	-	1.0000	1.0000
0.2	-	1.0000	1.0000
0.22	-	1.0000	1.0000
0.24	-	1.0000	1.0000
0.26	-	1.0000	1.0000
0.28	-	1.0000	1.0000
0.3	-	1.0000	1.0000
0.32	-	1.0000	1.0000
0.34	-	1.0000	1.0000
0.36	-	1.0000	1.0000
0.38	-	1.0000	1.0000
0.4	-	0.9565	0.9375
0.42	-	0.9565	0.9375
0.44	-	0.9565	0.9375
0.46	-	0.9565	0.9375
0.48	-	0.9167	0.8750
0.5	-	0.9167	0.8750
0.52	-	0.9167	0.8750
0.54	-	0.9167	0.8750
0.56	-	0.9167	0.8750
0.58	-	0.9167	0.8750
0.6	-	0.9167	0.8750
0.62	-	0.9167	0.8750
0.64	-	0.9167	0.8750
0.66	-	0.9167	0.8750
0.68	-	0.9167	0.8750
0.7	-	0.9167	0.8750
0.72	-	0.9167	0.8750
0.74	-	0.9167	0.8750
0.76	-	0.8800	0.8125
0.78	-	0.8800	0.8125
0.8	-	0.8800	0.8125
0.82	-	0.8800	0.8125
0.84	-	0.8800	0.8125
0.86	-	0.8800	0.8125
0.88	-	0.8800	0.8125
0.9	-	0.8800	0.8125
0.92	-	0.8800	0.8125
0.94	-	0.8800	0.8125
0.96	-	0.8800	0.8125
0.98	-	0.8800	0.8125
1	-	0.8800	0.8125

U_{DG} [1]	U_{DS} [2]	F_1 [3]	AC [4]
-	0.02	0.8148	0.6875
-	0.04	0.8148	0.6875
-	0.06	0.7200	0.5625
-	0.08	0.7200	0.5625
-	0.1	0.6667	0.5000
-	0.12	0.6087	0.4375
-	0.14	0.6364	0.5000
-	0.16	0.6364	0.5000
-	0.18	0.5714	0.4375
-	0.2	0.6000	0.5000
-	0.22	0.6000	0.5000
-	0.24	0.6000	0.5000
-	0.26	0.6000	0.5000
-	0.28	0.6000	0.5000
-	0.3	0.4444	0.3750
-	0.32	0.3529	0.3125
-	0.34	0.3529	0.3125
-	0.36	0.3750	0.3750
-	0.38	0.2667	0.3125
-	0.4	0.2667	0.3125
-	0.42	0.2667	0.3125
-	0.44	0.2667	0.3125
-	0.46	0.2857	0.3750
-	0.48	0.2857	0.3750
-	0.5	0.2857	0.3750
-	0.52	0.2857	0.3750
-	0.54	0.2857	0.3750
-	0.56	0.2857	0.3750
-	0.58	0.2857	0.3750
-	0.6	0.2857	0.3750
-	0.62	0.2857	0.3750
-	0.64	0.2857	0.3750
-	0.66	0.2857	0.3750
-	0.68	0.2857	0.3750
-	0.7	0.3077	0.4375
-	0.72	0.1667	0.3750
-	0.74	0.1667	0.3750
-	0.76	0.1667	0.3750
-	0.78	-	0.3125
-	0.8	-	0.3125
-	0.82	-	0.3125
-	0.84	-	0.3125
-	0.86	-	0.3125
-	0.88	-	0.3125
-	0.9	-	0.3125
-	0.92	-	0.3125
-	0.94	-	0.3125
-	0.96	-	0.3125
-	0.98	-	0.3125
-	1	-	0.3125

U_{DG} [1]	U_{DS} [2]	F_1 [3]	AC [4]
0.05	1	-	0.3125
0.05	0.9	-	0.3125
0.05	0.8	-	0.3125
0.1	0.95	-	0.3125
0.1	0.85	-	0.3125
0.1	0.75	0.1667	0.3750
0.15	0.9	-	0.3125
0.15	0.8	-	0.3125
0.15	0.7	0.3077	0.4375
0.2	0.85	-	0.3125
0.2	0.75	0.1667	0.3750
0.2	0.65	0.3077	0.4375
0.25	0.8	-	0.3125
0.25	0.7	0.3077	0.4375
0.25	0.6	0.3077	0.4375
0.3	0.75	0.1667	0.3750
0.3	0.65	0.3077	0.4375
0.3	0.55	0.3077	0.4375
0.35	0.7	0.3077	0.4375
0.35	0.6	0.3077	0.4375
0.35	0.5	0.3077	0.4375
0.4	0.65	0.3077	0.4375
0.4	0.55	0.3077	0.4375
0.4	0.45	0.3077	0.4375
0.45	0.6	0.3077	0.4375
0.45	0.5	0.3077	0.4375
0.45	0.4	0.3077	0.4375
0.5	0.55	0.2857	0.3750
0.5	0.45	0.2857	0.3750
0.5	0.35	0.4000	0.4375
0.55	0.5	0.2857	0.3750
0.55	0.4	0.2857	0.3750
0.55	0.3	0.5000	0.5000
0.6	0.45	0.2857	0.3750
0.6	0.35	0.4000	0.4375
0.6	0.25	0.6667	0.6250
0.65	0.4	0.2857	0.3750
0.65	0.3	0.5000	0.5000
0.65	0.2	0.6667	0.6250
0.7	0.35	0.4000	0.4375
0.7	0.25	0.6667	0.6250
0.7	0.15	0.7368	0.6875
0.75	0.3	0.4706	0.4375
0.75	0.2	0.6316	0.5625
0.75	0.1	0.7273	0.6250
0.8	0.25	0.6316	0.5625
0.8	0.15	0.7000	0.6250
0.8	0.05	0.8333	0.7500
0.85	0.2	0.6316	0.5625
0.85	0.1	0.7273	0.6250

1 - $\mathbf{U_{DG}}$ - Umbral de distancia del gesto detectado
2 - $\mathbf{U_{GDS}}$ - Umbral de distancia entre el gesto detectado y el segundo
3 - $\mathbf{F_1}$ - Valor F-1
4 - **AC** - Acierto

8.3.6. Comparación de Perfiles de Configuración

En la siguiente tabla se resumen los resultados de las 3 evaluaciones realizadas con los datos de **Acierto** (AC) y valor $\mathbf{F_1}$, para los 5 perfiles de configuración más importantes:

- D - Por Defecto. Sin preprocesado.
- C - Centrado.
- CR-B - Centrado y Rotado mediante cambio de base.
- CR-G - Centrado y Rotado mediante giro.
- CR-U - Centrado y Rotado. Distancia Selectiva con Umbral

11 GESTOS

PC[1]	$\mathbf{U_{DG}}$[2]	$\mathbf{U_{DS}}$[3]	**AC**[4]	$\mathbf{F_1}$[5]
D	-	-	73 %	0,81
C	-	-	**100 %**	**1**
CR-B	1	-	**100 %**	**1**
CR-G	1	-	**100 %**	**1**
CR-U	0,1	-	**100 %**	**1**

11 GESTOS GIRADOS

PC[1]	$\mathbf{U_{DG}}$[2]	$\mathbf{U_{DS}}$[3]	**AC**[4]	$\mathbf{F_1}$[5]
D	-	-	27 %	0,43
C	-	-	36 %	0,53
CR-B	1	-	**91 %**	**0.95**
CR-G	1	-	64 %	0,75
CR-U	0,1	-	**91 %**	**0.95**

16 GESTOS

PC[1]	$\mathbf{U_{DG}}$[2]	$\mathbf{U_{DS}}$[3]	**AC**[4]	$\mathbf{F_1}$[5]
D	-	-	69 %	0,81
C	-	-	69 %	0,81
CR-C	1	-	**94 %**	**0.96**
CR-G	1	-	**94 %**	**0.96**
CR-U	0,1	-	**100 %**	**1**

Cuadro 8.92: Comparativa de perfiles por evaluación

[1] **PC** - Perfil de Configuración
[2] $\mathbf{U_{DG}}$ - Umbral de distancia del gesto detectado
[3] $\mathbf{U_{GDS}}$ - Umbral de distancia entre el gesto detectado y el segundo
[4] **AC** - Porcentaje de Aciertos
[5] $\mathbf{F_1}$ - Valor F_1

El preprocesado que mejores resultados proprociona es **Centrado y Rotado mediante cambio de base**. Con este preprocesado tenemos 2 perfiles de configuración:

1. **CR-C** *Centrado y Rotado mediante cambio de base. Distancia selectiva.*
2. **CR-U** *Centrado y Rotado mediante cambio de base. Distancia selectiva con umbral.*

En el primer perfil de configuración, se utiliza una distancia del tipo **Selectiva**. En el segundo perfil, se utiliza una distancia del tipo **Selectiva con Umbral**. Comparando las 3 evaluaciones realizadas con cada perfil:

	CR-C $U_{DG} = 1$		**CR-U $U_{DG} = 0.1$**	
EV[1]	**ACC**[2]	**F_1**[3]	**ACC**[2]	**F_1**[3]
11	100 %	100 %	**100 %**	**100 %**
11G	91 %	95 %	**91 %**	**95 %**
16	94 %	96 %	**100 %**	**100 %**

Cuadro 8.93: Comparativa de los mejores perfiles de configuración

Aunque los resultados son muy buenos en los dos casos, el perfil con la distancia selectiva con umbral obtiene mejores resultados. El umbral $\mathbf{U_{DG}}$ puede variar desde 0,1 hasta 1 sin que apenas varíen los resultados.

[1] **EV** - Evaluación
[2] **ACC** - Acierto
[3] **F_1** - Valor F_1

Capítulo 9

Conclusión

En el capítulo anterior, hemos visto que el preprocesado de **centrado** es imprescindible, ya que compara los gestos como si la persona siempre estuviera situada en el mismo punto.

El preprocesado de **rotado** no es imprescindible si realizamos los gestos totalmente de frente al Kinect, pero en la realidad observamos que esto nunca ocurre, y nunca se realizan los gestos totalmente de frente al Kinect. Al realizar este preprocesado, situamos el cuerpo siempre de frente al Kinect, con lo que los resultados de acierto mejoran notablemente.

El preprocesado de rotado tiene 2 variantes: **rotado mediante cambio de base** y **rotado mediante giro sobre el eje Y**. Los evaluaciones realizadas nos muestran que los mejores resultados se obtienen con el primero de ellos: rotado mediante cambio de base.

Por tanto, la mejor configuración respecto a preprocesado se obtiene con centrado y rotado mediante cambio de base.

Otro aspecto importante es el tipo de distancia utilizado. En las evaluaciones hemos comparado los dos mejores tipos de distancia: **distancia selectiva** y **distancia selectiva con umbral**. Aunque con ambos se han obtenido muy buenos resultados, el último tipo de distancia, distancia selectiva con umbral, se han obtenido unos resultados ligeramente mejores.

Y por último, hemos visto que los resultados mejoran bastante estableciendo

un umbral de rechazo. Hay dos tipos de umbral de rechazo: $\mathbf{U_{DG}}$ (Umbral de rechazo teniendo en cuenta la distancia del gesto detectado) y $\mathbf{U_{GDS}}$ (Umbral de rechazo teniendo en cuenta la distancia entre el gesto detectado y el segundo).

El primero funciona muy bien, ya que los gestos fallados suelen tener una distancia mucho mayor. En la evaluación de 16 gestos para el perfil centrado y rotado mediante cambio de base se mejora el Accuracy del 69 % sin umbral de rechazo, al 100 % con umbral de rechazo.

El segundo no funciona tan bien, ya que la distancia entre el gesto detectado y el segundo de los gestos acertados no es en general mayor o menor que la de los gestos fallados.

Por tanto, concluimos que hay **4 elementos imprescindibles** que debemos habilitar en la configuración si queremos obtener los mejores resultados:

1. **Centrado**
2. **Rotado mediante Cambio de Base**
3. **Umbral de rechazo $\mathbf{U_{DG}}$**
4. **Distancia selectiva** (con o sin umbral)

La diferencia entre la distancia selectiva o selectiva con umbral no es muy relevante, aunque dependiendo de cual escojamos tendremos que configurar el umbral de rechazo, $\mathbf{U_{DG}}$, con un valor u otro. En el caso de distancia selectiva, un valor adecuado de $\mathbf{U_{DG}}$ es 1. Con distancia selectiva con umbral, un valor adecuado de $\mathbf{U_{DG}}$ es 0,1.

Con estas configuraciones, en las evaluaciones realizadas se han obtenido los siguientes resultados:

EV.	ACIERTO	VALOR $\mathbf{F_1}$
11	**100 %**	**100 %**
11G	**91 %**	**95 %**
16	**100 %**	**100 %**

Resultados para **centrado** y **rotado mediante Cambio de Base**

Distancia **selectiva** con umbral. $\mathbf{U_{DG}} = 0.1$

Parte II

Desarrollo de un sistema de teleoperación de robots integrando el reconocedor de gestos realizado

Capítulo 10

Estado del Arte

A través de este capítulo, veremos el uso del reconocimiento de gestos para teleoperar sistemas y su integración en robots autónomos. En la teleoperación de sistemas, el reconocimiento de gestos es un sustituto en muchos casos de los mandos tradicionales; y en los robots autónomos, el reconocimiento de gestos es una habilidad totalmente necesaria para la interacción con el ser humano de una manera natural.

10.1. Teleoperación de sistemas mediante el reconocimiento de gestos

El reconocimiento de gestos aplicado a la teleoperación de sistemas se está extendiendo en los últimos años. Gracias a la madurez de algunas técnicas de reconocimiento de gestos, y la aparición de sistemas de bajo coste capaces de detectar imágenes en 3 dimensiones como Kinect, hoy en día es posible manejar sistemas con sólo realizar gestos.

10.1.1. Control de televisión - Pointgrab

Pointgrab es una compañía que ha desarrollado un sistema para controlar la televisión, basado en el reconocimiento de gestos con las manos usando una

cámara 2D. Según su descripción, es capaz de detectar y seguir la mano hasta 5 metros con diversas condiciones de luz, y puede ser integrado en cualquier televisión inteligente.

10.1.2. Control de videconsolas - Kinect

Aunque Kinect se utilice en este proyecto para reconocer gestos en un ordenador, en realidad Kinect fue diseñado como un controlador para videoconsolas. Gracias al reconocimiento visual, ya no es necesario usar un mando en la Xbox 360.

10.1.3. Control del ordenador - Kinect for Windows

Tras el éxito de Kinect en su uso para el reconocimiento de gestos en el ordenador, Microsoft decidió lanzar una versión específica para ordenador, llamada *Kinect for Windows*. El hardware es el mismo, excepto que el sensor de profundidad mejora el rango cercano y es capaz de detectar objetos desde los 40 cm en vez de desde los 80 cm que es capaz de detectar el Kinect para Xbox.

Con la aparición de *Kinect for Windows* se han creado programas que permiten usar el Kinect para realizar ciertos movimientos, como pasar diapositivas o pasar páginas.

10.2. Reconocimiento de gestos en robots autónomos

El reconocimiento de gestos o voz no se limita sólo a la teleoperación de robots, sino que también puede servir para interactuar con un robot autónomo. A continuación veremos distintos ejemplos de robots autónomos, y cómo el reconocimiento de gestos es imprescindible para la interacción con el ser humano.

10.2.1. Asistentes - Robonauta

Los robots asistentes son sistemas inteligentes diseñados para ayudar a personas. Este tipo de ayudas puede ser muy variado, como guiar a personas en un museo, indicar dónde está un determinado lugar, informar sobre un asunto... En todas estas situaciones, los gestos forman parte de la comunicación de las personas, y los sistemas inteligentes necesitan entender estos gestos para hacer la interacción con las personas totalmente natural.

Figura 10.1: Robonauta - Robot Asistente

Fuente: Canal GTH Youtube (http://www.youtube.com/user/robonautaGTH)

El proyecto Robonauta del Grupo de Tecnología del Habla, es un robot que incorpora el reconocimiento del habla para interactuar con personas. Robonauta es *un sistema inteligente de asistencia para el acceso personalizado a lugares públicos, y una interacción fluida con la información y con otros ciudadanos allí presentes.*

Mediante el reconocimiento de gestos, un sistema inteligente como el robonauta sería capaz de interpretar gestos, como un saludo o una indicación. De esta manera el sistema inteligente daría un paso más en la interacción con el ser humano, ya que sería capaz de interactuar en con personas mediante gestos.

10.2.2. Coches autónomos - Junior

Los coches autónomos son sistemas inteligentes diseñados para conducir de forma autónomona. El primer coche autónomo fue creado en 1958 mediante la colaboración de General Motors y RCA Labs, y desde entonces han sido muchas compañías las que han fabricado sus propios modelos: Mercedes-Benz, General Motors, Nissan, Toyota, Audi... incluso empresas del sector tecnológico como Google.

A día de hoy los coches autónomos sólo se han desarrollado como modelos de investigación, aunque algunas de las tecnologías de los coches autónomos han acabado por ser introducidas en los coches de venta al público.

Figura 10.2: Junior - Coche Autonomo

Fuente: Stanford University

Junior es un Volkswagen Passat que integra el sistema *autonomous valet parking*, gracias al cual el conductor puede salir del coche y decirle al vehículo que aparque solo. Además, incorpora un sistema mediante el cual el propietario puede llamar al coche mediante una aplicación del móvil y el coche va sólo hasta donde esté el propietario.

En la conducción, el reconocimiento de gestos es totalmente necesario. Imaginemos un escenario típico en el hay un guardia de tráfico desviando el tráfico hacia una determinada calle. Si el vehículo autónomo es incapaz de reconocer gestos, ignoraría las órdenes del guardia de tráfico y podría provocar un accidente.

10.2.3. Robots humanoides - Robot Violinista de Toyota

Los robots humanoides son robots cuyo cuerpo está diseñado para que parezca una persona. Estos robots son diseñados para imitar el comportamiento humano en alguna faceta, como el robot *TOPIO* (Ping Pong Playing Robot), que juega al ping-pong o el robot violinista de Toyota.

Figura 10.3: Robot Violinista de Toyota

Fuente: Toyota Pressroom (http://corporatenews.pressroom.toyota.com/)

Si se integra el robot violinista en una orquesta, es necesario que el robot violinista interprete los gestos del director de orquesta para seguir sus indicaciones de tiempo, entradas... Sin el reconocimiento de gestos, un robot músico no podría estar integrado en una orquesta.

Capítulo 11

Lego Mindstorm EV3

Lego Mindstorm es un kit de robótica diseñado para crear y programar un robot de una manera muy sencilla. Incluye una unidad central programable que controla el sistema, un conjunto de motores y sensores modulares, y piezas lego para construir el robot.

Figura 11.1: Lego Mindstorm EV3

El Lego Mindsotrm EV3 es la tercera generación de la serie de robots LEGO Mindstorm, sucesor de la segunda generación de robots Lego Minstorm NXT.

Las características técnicas del Lego Mindstorm EV3 son:

La unidad central del Lego Mindsotrm es donde está ubicado el procesador, la memoria y los puertos de entrada y de salida.

Pantalla	LCD Monocromo de 178 x 128 píxeles
Procesador	300 MHz (ARM9)
Memoria	64 MB RAM, 16 MB Flash
Sistema Operativo	Linux
Blueetooth	Si
USB	Si

Cuadro 11.1: Características del Lego Mindstorm EV3

Figura 11.2: Unidad Central del Lego Mindstorm

11.1. Motores

El robot Lego Mindstorm utilizado en este proyecto tiene 4 servomotores:

- Brazo izquierdo - Motor A
- Brazo derecho - Motor C
- Pierna izquierda - Motor B
- Pierna derecha - Motor D

Gracias a la librería *Monobrick*, es posible enviar comandos al robot Mindstorm con C# mediante USB o WIFI. Los movimiento de brazos o piernas son bastante sencillos: basta con activar el motor con una velocidad durante un tiempo.

Para los brazos, las velocidades con valores positivos bajan el brazo, y con valores positivos los levantan.

Para las piernas, las velocidades con valores negativos adelantan la pierna, y con valores positivos atrasan la pierna.

Por ejemplo, para levantar el brazo izquierdo, activaremos el motor A, con velocidad -50, durante 2 segundos:

```
string connection = usb;
Brick<Sensor, Sensor, Sensor, Sensor>v3 = new
   Brick<Sensor, Sensor, Sensor, Sensor>(connection);
ev3.Connection.Open();
ev3.MotorA.On(-50);
Thread.Sleep(2000);
ev3.MotorA.Off();
```

Para atrasar la pierna derecha, activaremos el motor D, con velocidad 50 durante 2 segundos:

```
ev3.MotorD.On(50);
Thread.Sleep(2000);
ev3.MotorD.Off();
```

Si queremos activar dos motores a la vez, debemos usar la opción de *MotorSync*. Por ejemplo, para mover ambos pies hacia adelante:

```
ev3.Connection.Open();
ev3.MotorSync.BitField = OutputBitfield.OutB |
   OutputBitfield.OutD;
ev3.MotorSync.On(50, 0);
Thread.Sleep(2000);
ev3.MotorSync.Off();
```

Capítulo 12

Software: Teleoperador de Robot Mindstorm

El Teleoperador de Robot Mindstorm es el *sistema de teleoperación de robots basado en reconocimiento de gestos usando el sensor Kinect*. Al igual que el *Evaluador de Gestos*, el *Teleoperador de Robot Mindstorm* hace uso de la librería *Algorithms* para la detección de gestos, y de la base de datos para guardar toda la configuración.

En este capítulo veremos el sistema de teleoperación de robots, la integración con la librería y la base de datos, y las nuevas funcionalidades que se han añadido a la librería para el reconocimiento de voz y a la base de datos para guardar la configuración relativa al teleoperador de Robot Mindstorm.

Estructura La solución del *Teleoperador de Robot Mindstorm* de gestos está dividida en 3 proyectos:

- *Algorithms* - Librería interna que sirve de soporte al Controlador de Robot Mindstorm.
- *MindstormController* - Controlador de Robot Mindstorm. Hace uso de la librería interna (Algorithms) y de la base de datos (Models).
- *Models* - Base de datos.

12.1. Algorithms

12.1.1. Reconocedor de Audio - *SpeechRecognizer*

El reconocimiento de voz implementado para controlar el robot Lego Mindstorm utiliza el motor de reconocimiento de Windows. Para ello se hace uso de la plataforma *Microsft Speech Platform*.

Existen diferentes motores de reconocimiento para cada idioma. Para esta aplicación se ha utilizado el motor de reconocimiento de español y el de inglés. Dependiendo el idioma en que se está mostrando la aplicación, se cargará uno u otro motor de reconocimiento. En la página de *Microsft Speech Platform* existen motores para multitud de idiomas: francés, italiano, alemán, japonés...

El motor de reconocimiento necesita un archivo de gramática, que es un xml donde se le indica el idioma y las palabras que tiene que reconocer. El motor de reconocimiento al detectar una palabra, devuelve la etiqueta asociada a esa palabra. Una o varias palabras pueden estar asociadas a la misma etiqueta.

```
<grammar version="1.0" xml:lang="es-ES"
   root="rootRule" tag-format="semantics/1.0-literals"
   xmlns="http://www.w3.org/2001/06/grammar">
  <rule id="rootRule">
    <one-of>
      <item>
        <tag>GO FORWARD</tag>
        <one-of>
          <item> ir recto </item>
          <item> adelante </item>
        </one-of>
      </item>
                  </one-of>
  </rule>
</grammar>
```

En la aplicación se han incorporado 2 archivos de gramática: uno en español y otro inglés. Dependiendo el idioma en que se está mostrando la aplicación, se

cargará uno u otro fichero.

Podemos observar la etiqueta, *GO FORWARD*, con varias palabras asociadas: *ir recto*, *adelante*, *adelantar ambos pies*. Cuando se detecte cualquiera de estas palabras, el motor de reconocimiento devolverá como resultado la etiqueta *GO FORWARD*.

El resultado se acompaña del grado de confianza de que esa palabra sea la palabra buscada. La confianza es un valor entre 0 y 1. En la aplicación se ha configurado un umbral de confianza superior a 0.3 para que se considere una detección válida.

La clase *SpeechRecognizer* tiene las siguientes propiedades:

language	Idioma del reconocimiento.
sensor	Conector con el sensor kinect.
speechEngine	Motor de reconocimiento.
speechRecognizedEventHandler	Evento que se lanza cuando se detecta una frase.

La clase *SpeechRecognizer* tiene los siguientes métodos:

close	Para el reconocimiento de voz.
GetKinectRecognizer	Obtiene el motor de reconocimiento del idioma configurado.
start	Inicia el reconocimiento de voz.

12.2. MindstormController

12.2.1. Ventanas

Ventana Inicial - (*MainWindow*)

La ventana inicial es la pantalla que nos aparece nada más ejecutar la aplicación. Podemos acceder introduciendo usuario y contraseña, o podemos crear un nuevo usuario. También es posible cambiar el idioma de la aplicación entre español e inglés.

Figura 12.1: Pantalla de Inicio del MindstormController

La clase correspondiente de la ventana inicial es la clase *MainWindow*. Al crearse la clase, se comprueba en un thread aparte que hay comunicación con la base de datos y que la estructura de la misma corresponde con la estructura de la base de datos definida en la aplicación.

Si existe algún problema con la base de datos, bien porque hay un error de comunicación, de credenciales o el modelo de la base de datos no concuerda con el modelo actual (se ha modificado alguna tabla), la aplicación muestra un mensaje de error y no se puede acceder a la aplicación hasta que se solucione el problema.

La clase *MainWindow* contiene las siguientes propiedades:

bw	BackgroundWorker. Ejecuta procesos de fondo en otro thread.
ctx	Conexión con la base de datos.

La clase *MainWindow* contiene los siguientes métodos:

Button_Click	Comprueba las credenciales del usuario, y si son correctos, va a la Pantalla Principal.
checkDatabase	Comprueba si hay conexión a la base de datos. Si no hay conexión, muestra un mensaje en pantalla.
EnglishClick	Cambia el idioma a inglés
Exit_Click	Se sale de la aplicación
isDtabaseUp	Devuelve true si hay conexión a la base de datos.
MainWindow	Constructor
NewUserClick	Se va a la pantalla de crear un nuevo usuario.
SetLanguageDictionary	Cambia al idioma especificado en el argumento.
SpanishClick	Cambia el idioma a español

Registro de Usuarios - (*RegisterWindow*)

El reconocimiento de gestos es un proceso que requiere de un entrenamiento de gestos previo. Aunque teóricamente se podría usar un sólo entrenamiento para reconocer los gestos realizados por cualquier persona, en la práctica observamos que cada persona no realiza los mismo gestos de la misma manera. Por ejemplo, el gesto de levantar la mano se puede realizar desde una altura hasta otra altura, y a distintas velocidades.

Por este motivo, el evaluador de gestos tiene un sistema de usuarios donde cada usuario realiza su propio entrenamiento y puede realizar sus propias evaluaciones.

El registro de usuario es muy simple, y tiene los campos de Nombre, Contraseña e Idioma:

- El **Nombre** es un elemento único, por lo que no puede haber 2 usuarios con el mismo nombre. Es similar al Login de un sistema operativo, correo

o cualquier aplicación, que es único.

- La **Contraseña** puede ser cualquier combinación de caracteres. Por motivos de seguridad, en la base de datos se guarda simplemente el hash de la contraseña.
- El **Idioma** permite mostrar la aplicación para el usuario por defecto en Inglés o en Español. Realizar una traducción a otro idioma es muy sencillo, ya que sólo hay que traducir el fichero StringResources.xaml.

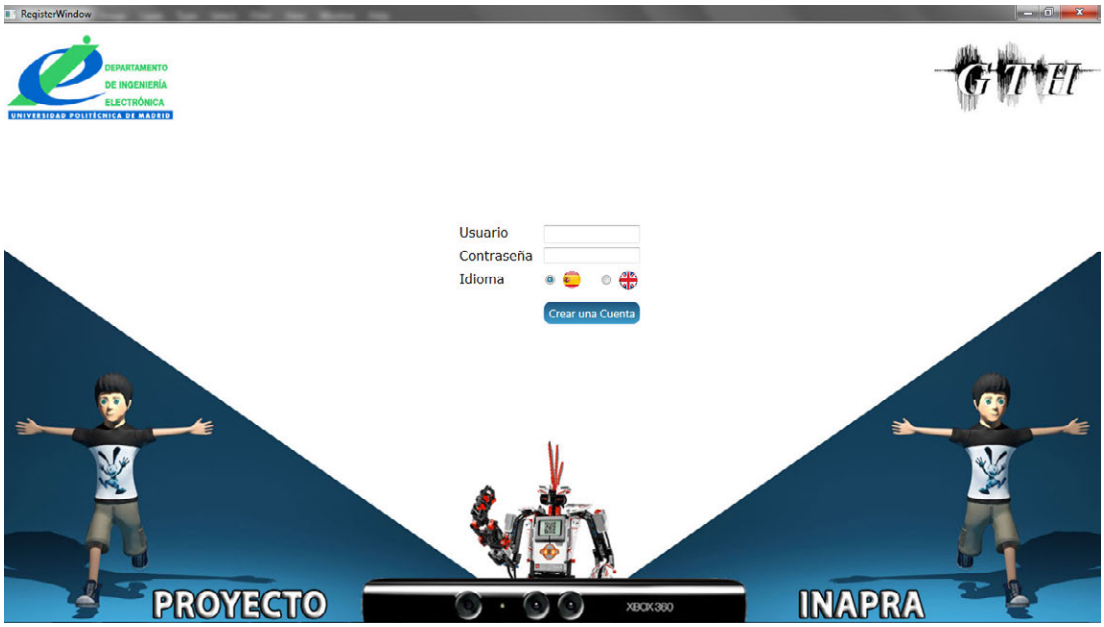


Figura 12.2: Pantalla de Registro de Usuarios

La clase *RegisterWindow* contiene las siguientes propiedades:

ctx	Conexión con la base de datos.

La clase *RegisterWindow* contiene los siguientes métodos:

Register_Click	Crea un nuevo Usuario en la base de datos con los campos introducidos: usuario, contraseña e idioma. Si algún campo está vacío, se muestra un error en pantalla y no se crea el usuario en la base de datos.
RegisterWindow	Constructor.

Ventana de navegación Principal - (*MainWindow*)

La clase *HomeWindow* es una ventana de navegación (*NavigationWindow*) que hace de contenedor para las distintas páginas. En esta ventana se irán cargando las distintas páginas, incluida la propia página principal (*HomePage*).

La clase *HomeWindow* contiene las siguientes propiedades:

username	Nombre del usuario que se ha autenticado. Esta propiedad es una propiedad estática y es utilizado en los accesos a base de datos para especificar el usuario. En una página web sería el equivalente a una variable de sesión.

La clase *HomeWindow* contiene los siguientes métodos:

HomeWindow	Constructor.

12.2.2. Páginas

Página principal - (*HomePage*)

Una vez introducidos el usuario y contraseña, se accede a la Home. Desde esta pantalla se puede ir a *Entrenamiento* para realizar el entrenamiento de gestos, o *Detección*, para realizar una evaluación o detectar gestos continuamente.

Desde la Home se pueden realizar las siguientes acciones:

- Ir a la página de Entrenamiento
- Ir a la página de Detección
- Cerrar Sesión, haciendo click en el botón *Cerrar Sesión*
- Salir de la aplicación, haciendo click en el botón *Salir*
- Cambiar de idioma, haciendo click en el botón con la bandera español (Español) o en el botón con la bandera inglesa (Inglés).

Figura 12.3: Home del MindstormController

La clase *HomePage* contiene las siguientes propiedades:

ctx	Conexión con la base de datos.

La clase *HomePage* contiene los siguientes métodos:

CloseSession_Click	Cierra la sesión. La aplicación va a la pantalla principal.
English_Click	Cambia el idioma a inglés
Evaluation_Click	Se va a la página de Evaluación.
Exit_Click	Se sale de la aplicación.
HomePage	Constructor.
SetLanguageDictionary	Cambia al idioma especificado en el argumento.
Spanish_Click	Cambia el idioma a español
Training_Click	Se va a la página de Entrenamiento.

Configuración - (*ConfigurationPage*)

La configuración del teleoperador de Robot Mindstorm permite configurar los movimientos del robot, el tipo de conexión, el perfil de detección de gestos y el método de reconocimiento.

Figura 12.4: ConfigurationPage

La configuración de movimiento permite asignar gestos a los distintos movimientos que puede realizar el robot. De esta manera, cuando se detecta un gesto, el robot realizará la acción configurada. Las acciones que puede realizar el robot son las siguientes:

- Mover brazo derecho: si el brazo está abajo, lo sube; si está arriba, lo baja.
- Levantar brazo derecho.
- Bajar brazo derecho.
- Mover brazo izquierdo: si el brazo está abajo, lo sube; si está arriba, lo baja.
- Levantar brazo izquierdo.
- Bajar brazo izquierdo.
- Mover ambos brazos: si brazos están abajo, los sube; si están arriba, los

baja.

- Levantar ambos brazos.
- Bajar ambos brazos.
- Mover pie derecho: si el pie está delante, lo atrasa; si está atrás, lo adelanta.
- Adelantar pie derecho.
- Atrasar pie derecho.
- Mover pie izquierdo: si el pie está delante, lo atrasa; si está atrás, lo adelanta.
- Adelantar pie izquierdo.
- Atrasar pie izquierdo.
- Mover ambos pies: si los pies están delante, los atrasa; si están atrás, los adelanta.
- Adelantar ambos pies.
- Atrasar ambos pies.

El tipo de conexión con el robot Lego Mindstorm puede ser **USB** o **WIFI**. Para un uso continuo, se recomienda configurar la conexión USB, ya que el robot Lego Mindstorm no gastará batería interna (6 pilas en serie). La conexión WIFI se recomienda para presentaciones, donde no se verán cables por medio. Los vídeos que se han grabado en este proyecto (Ver Anexo - Pág. 259) han sido grabados usando la conexión WIFI.

El perfil de configuración para la detección de gestos establece los umbrales de energía y tiempo y el preprocesamiento a la hora de detectar el movimiento.

El tipo de reconocimiento puede ser audio o visual. **Audio** indica que se reconocerán órdenes por voz asociadas a los movimientos del robot. **Visual** indica que se reconocerán gestos asociados a los movimientos del robot.

Teleoperador del Robot Mindstorm - (*RobotControllerPage*)

El teleoperador del robot Mindstorm permite controlar el robot Mindstorm mediante reconocimiento de voz o reconocimiento de gestos, dependiendo de la opción seleccionada en la configuración.

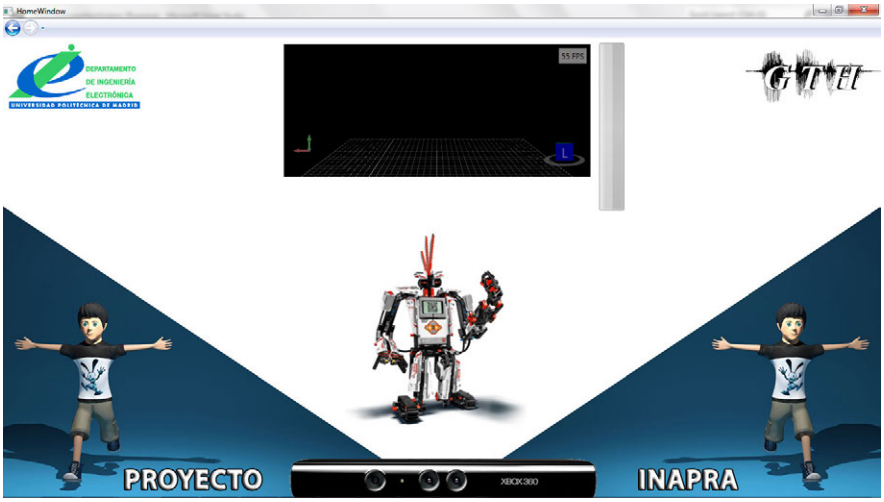


Figura 12.5: Teleoperador del Robot Mindstorm

Las acciones que puede realizar el robot son las vistas en la sección de configuración.

La clase *RobotControllerPage* contiene las siguientes propiedades:

bw1	Realiza procesos en otro thread.
conf	Configuración
ctx	Conector con la base de datos.
enableRecord	Habilita la grabación de un gesto.
ev3	Conector con el robot Mindstorm.
gesture	Gesto actual.
gestureRecognizer	Reconocedor de gestos.
leftArm	Estado del brazo izquierdo (Arriba o Abajo).
leftFoot	Estado del pie izquierdo (Atrás o Adelante).
rightArm	Estado del brazo derecho (Arriba o Abajo).
rightFoot	Estado del pie derecho (Atrás o Adelante).
sensor	Conector con el Kinect.
speechRecognizer	Reconocedor de Voz.

La clase *RobotControllerPage* contiene los siguientes métodos:

BothArmsDown	Baja ambas manos.
BothArmsDownCommand	Mueve motores para bajar ambas manos.
BothArmsMove	Mueve ambas manos.
BothArmsMoveCommand	Mueve motores para mover ambas manos.
BothArmsUp	Sube ambas manos.
BothArmsUpCommand	Mueve motores para subir ambas manos.
BothFeetBackward	Atrasa ambos pies.
BothFeetBackwardCommand	Mueve motores para atrasar ambos pies.
BothFeetForward	Adelanta ambos pies.
BothFeetForwardCommand	Mueve motores para atrasar ambos pies.
BothFeetMove	Mueve ambos pies.
BothFeetMoveCommand	Mueve motores para mover ambos pies.
finishGesture	Métdo cuando termina un gesto.
Move	Mueve un motor.
MoveDual	Mueve dos motores síncronamente.
SendCommand	Envía un comando al robot.
SendMoveCommand	Envía un comando al robot de tipo Move (si el motor ha girado antes en un sentido, ahora gira en el otro).
sensor_SkeletonFrameReadyRecord	Pinta los esqueletos en la pantalla.
SpeechRecognized	Atiende los eventos de reconocimiento de voz.
startGestureRecognizer	Ejecuta el reconocedor de gestos.
WaitForMotorToStop	Espera hasta que el motor pare.
WriteText	Escribe texto en la pantalla.

12.2.3. Extensiones

Estados del Motor - (*MotorState*)

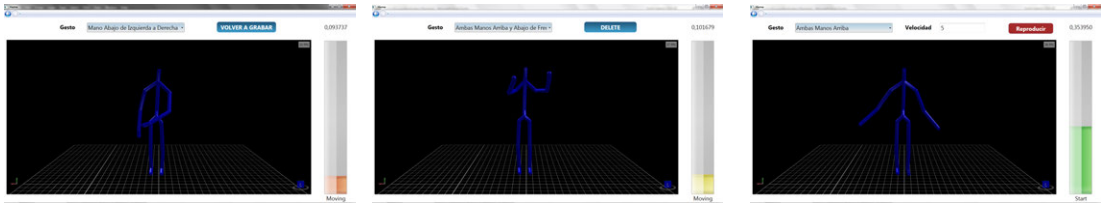
Para realizar un movimiento con las manos, se activa el motor correspondiente del robot lego mindstorm durante un tiempo determinado. Por ejemplo, para levantar la mano derecha, se activa el motor C durante 1,8 segundos.

Para hacer más fácil la interación con el robot Mindstorm, se ha definido sólo 2 estados: *Arriba* o *Abajo*. El enumerado *MotorState* define esos estados como Down (Abajo) y Arriba (Up).

Escala de Colores - (*RedToGreenScaleConverter*)

La clase *RedToGreenScaleConverter* define una escala de colores para la barra de energía cuando se realiza un gesto. Esta barra de energía indica según cada color si ha superado el primer umbral de energía (reposo a inicio de movimiento) y el segundo umbral de energía (inicio de movimiento a movimiento).

- Rojo: No ha superado el primer umbral de energía.
- Amarillo: Ha superado el primer umbral de energía, pero no el segundo.
- Verde: Ha superado el segundo umbral de energía.



(a) Rojo (b) Amarillo (c) Verde

Figura 12.6: Umbrales de Energía

12.3. Models

12.3.1. Configuración

MindstormConfiguration

En esta tabla se almacena la configuración relativa al controlador de robot Mindstorm.

- **ID** - Identificador único de la configuración del preprocesamiento del esqueleto. Clave primaria autoincremental.
- **user** - Usuario al que pertenece la configuración.
- **connection** - Tipo de conexión con el robot Mindstorm: USB o WIFI.
- **recognition** - Tipo de reconocimiento: voz o gestos.
- **profile** - Perfil de configuración.
- **rightArmMove** - Gesto asociado con el movimiento del brazo derecho. Si el brazo está arriba, se mueve abajo. Si estaba abajo, se mueve arriba.
- **rightArmUp** - Gesto asociado con el movimiento del brazo derecho arriba.
- **rightArmDown** - Gesto asociado con el movimiento del brazo derecho abajo.
- **leftArmMove** - Gesto asociado con el movimiento del brazo izquierdo. Si el brazo está arriba, se mueve abajo. Si estaba abajo, se mueve arriba.
- **leftArmUp** - Gesto asociado con el movimiento del brazo izquierdo arriba.
- **leftArmDown** - Gesto asociado con el movimiento del brazo izquierdo abajo.
- **bothArmsMove** - Gesto asociado con el movimiento de ambos brazos. Si ambos brazos están arriba, se mueven abajo. Si estaban abajo, se mueven arriba.
- **bothArmsUp** - Gesto asociado con el movimiento de ambos brazos arriba.

- **bothArmsDown** - Gesto asociado con el movimiento de ambos brazos abajo.
- **rightFootMove** - Gesto asociado con el movimiento del pie derecho. Si el pie está detrás, se mueve adelante. Si estaba adelante, se mueve atrás.
- **rightFootForward** - Gesto asociado con el movimiento del pie derecho adelante.
- **rightFootBackward** - Gesto asociado con el movimiento del pie derecho atrás.
- **leftFootMove** - Gesto asociado con el movimiento del pie izquierdo. Si el pie está detrás, se mueve adelante. Si estaba adelante, se mueve atrás.
- **rightFootBackward** - Gesto asociado con el movimiento del pie derecho atrás.
- **leftFootForward** - Gesto asociado con el movimiento del pie izquierdo adelante.
- **leftFootBackward** - Gesto asociado con el movimiento del pie izquierdo atrás.
- **bothFeetMove** - Gesto asociado con el movimiento de ambos pies. Si ambos pies están detrás, se mueven adelante. Si estaban adelante, se mueves atrás.
- **bothFeetForward** - Gesto asociado con el movimiento de ambos pies adelante.
- **bothFeetBackward** - Gesto asociado con el movimiento de ambos pies atrás.

Capítulo 13

Conclusiones del sistema final

EL sistema final de este proyecto es un *sistema de teleoperación de robots basado en reconocimiento de gestos usando el sensor Kinect*. Por tanto, las conclusiones del sistema final son las conclusiones del controlador de robot Mindstorm:

- El Kinect tiene un campo de visión limitado que tendremos que tener en cuenta. Para estar dentro del campo de visión, procuraremos situarnos entre 1 metro y 4 metros de distancia del Kinect y a no más de 2 metros a la derecha o la izquierda del Kinect. La mejor posición es justo en frente del Kinect, a una distancia de 2.5 metros.

- Cada persona realiza los movimientos de una manera: más rápidos o más lentos, con más recorrido o con menos recorrido. La mejor manera de adaptarse a cada persona es configurando los umbrales de energía y de tiempo. Como hay muchos umbrales (6 umbrales de tiempo y 6 de energía), realmente sólo es necesario variar 2 umbrales en dos casos:

    - Si la persona realiza los gestos lentamente y el sistema detecta falsos finales de movimiento, tendremos que disminuir el umbral de energía de Inicio de Movimiento a Movimiento. Movimientos lentos = menor energía de movimiento y mayor tiempo

    - Si la persona realiza los gestos muy rápido y no se detectan los movimiento como gestos, tendremos que disminuir el umbral de tiempo de Inicio de Movimiento a Movimiento. Movimiento rápidos = mayor energía de movimiento y menor tiempo

- El preprocesamiento del esqueleto de **centrado** y **rotado mediante cambio de base** ha mejorado notablemente el porcentaje de detección de gestos, pasando de un 64 % a un 100 % en la evaluación de 11 gestos.

- Los mejores resultados de **acierto** y valor $\mathbf{F_1}$ se obtienen con el **Umbral de rechazo teniendo en cuenta la distancia del gesto detectado**, $\mathbf{U_{DG}}$. En la evaluación de 16 gestos para el perfil centrado y rotado mediante cambio de base, se mejora el acierto del 69 % sin umbral de rechazo, al 100 % con umbral de rechazo $\mathbf{U_{DG}}$.

- Se recomienda que el perfil de configuración utilizado para calcular la energía del movimiento sea el perfil por defecto. Esto nos dará la energía real del movimiento. Si configuramos un perfil como *Centrado*, o *Centrado y Rotado - Distancia como Componentes al Cubo*, habrá que tener en cuenta que la energía se calculará tras el preprocesamiento, lo que dará lugar a la obtención de la energía del movimiento manipulada.

- El algoritmo DTW es un algoritmo muy robusto que funciona muy bien a la hora de alinear gestos. Sin embargo las variantes DTW no han mostrado ninguna mejor respecto al algoritmo original.

- El sistema de teleoperación de robot se ha mostrado más robusto mediante detección de gestos que mediante órdenes de voz. Con las pruebas realizadas, mediante detección de gestos se detectaron todos los gestos perfectamente excepto cuando la persona giraba 180º. Sin embargo, mediante reconocimiento de voz había órdenes que era necesario repetirlas.

- En condiciones normales (situados en frente del Kinect, a una distancia de entre 2 y 3 metros), se han realizado evaluaciones y demostraciones de controlar el robot Mindstorm perfectas (fiabilidad 100 %) usando 11 gestos y la configuración de *Centrado y Rotado mediante cambio de base*, y un umbral de rechazo $\mathbf{U_{DG}}$.

- Las evaluaciones realizadas con 11 gestos y 16 gestos (5 de ellos inexistentes), se han obtenido unos valores de **acierto** y **F1** igual al 100 %.

Parte III

Apéndice

Apéndice A

Videos demostrativos

Se han creado varios videos demostrativos de la aplicación que se han publicado en Youtube con el objetivo de mostrar el uso tanto del Evaluador de Gestos como del Teleoperador de Robot Mindstorm:

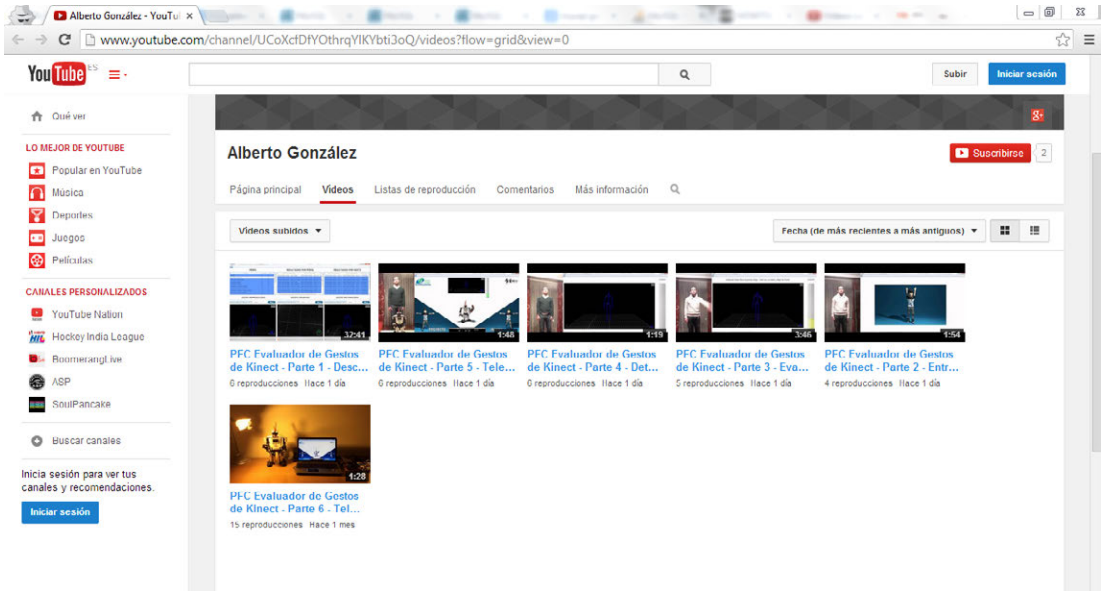


Figura A.1: Videos del Proyecto

1. **Evaluador de Gestos**

   http://www.youtube.com/watch?v=XRPatU02rZA

2. **Entrenamiento**

   http://www.youtube.com/watch?v=oX2K2C8cvbU

3. **Evaluación**

   http://www.youtube.com/watch?v=igx4bwwsFLs

4. **Detección de Modo Continuo**

   http://www.youtube.com/watch?v=7RNsVds8Op4

5. **Teleoperar el Robot Mindstorm mediante gestos**

   http://www.youtube.com/watch?v=ZKHVrhJwS0s

6. **Teleoperar el Robot Mindstorm mediante la voz**

   http://www.youtube.com/watch?v=AKNR4Weq6Mo

Figura A.2: Video del Evaluador de Gestos

Apéndice B

Importar y Exportar la Base de Datos del Evaluador de Gestos

El evaluador de gestos guarda toda la configuración en una base de datos. A continuación se van a explicar los pasos para importar o exportar la base de datos del Evaluador de Gestos.

B.1. Exportar la Base de Datos

Si estamos en un entorno Linux, basta con realizar el comando:

mysqldump -ugrs -pgesture gestures >gestures.sql

Si estamos en un entorno Windows, podemos utilizar una herramienta gráfica como *HeidiSQL* o *phpMyAdmin* que permiten la exportación de la base de datos.

B.2. Importar la Base de Datos

Los pasos para importar la base de datos *gestures.sql* son los siguientes:

1. Crearemos la base de datos *gestures*.

2. Crearemos el usuario *grs*

3. Daremos permisos al usuario grs sobre la base de datos con los siguientes comandos SQL:

   GRANT ALL PRIVILEGES ON 'gestures'. TO 'grs'@'localhost'WITH GRANT OPTION; FLUSH PRIVILEGES;*

4. Importaremos la base de datos

   En un entorno linux, usaremos el comando:

   mysql -ugrs -pgesture gestures <gestures.sql

   En un entorno Windows, podremos usar cualquier herramienta gráfica que nos permita importar una base de datos.

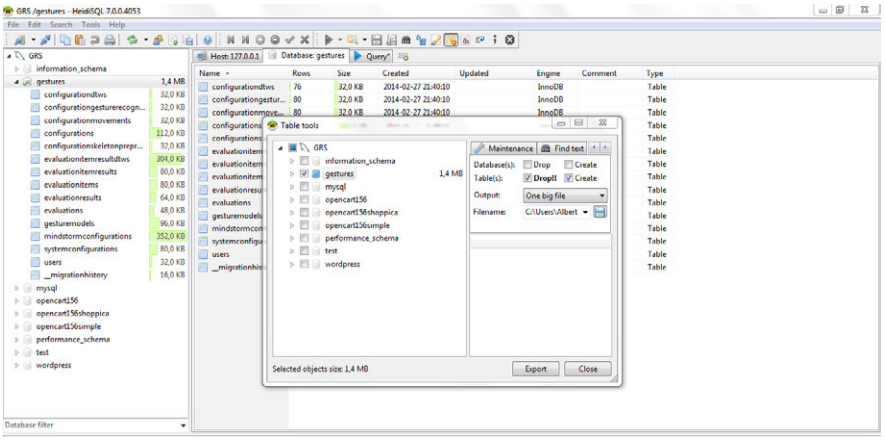


Figura B.1: Exportar Base de Datos

Apéndice C

Importar y Exportar gestos del Evaluador de Gestos

Los gestos se guardan en archivos de texto dentro de la carpeta *Data*, que está situada en el mismo directorio donde está el ejecutable del Evaluador de Gestos. En la base de datos, se guardan las rutas absolutas de los archivos de los gestos para que la aplicación puede ejecutarse desde cualquier directorio sin que haya problemas de rutas. Sin embargo, esto representa un problema cuando los gestos se importan, ya que la ruta donde se crearon esos gestos seguramente no exista donde se importen los gestos. Por esta razón se explicará en este apéndice como exportar e importar los gestos.

Supondremos que la aplicación Evaluador de Gestos donde se realizaron los gestos estaba instalada en `C:\Program Files\Evaluador de Gestos`, y que ahora vamos a importarla en otro ordenador donde la tenemos instalada en `C:\Users\Alberto \Documents\Evaluador de Gestos`

C.1. Exportar los Gestos

Exportar los gestos es un proceso muy sencillo. Basta copiar la carpeta Data y anotar el directorio donde está la carpeta Data. En nuestro ejemplo, `C:\Program Files\Evaluador de Gestos`.

C.2. Importar los Gestos

Para importar los gestos, necesitamos realizar dos pasos:

1. Situaremos los gestos exportados (directorio Data) en nuestro ordenador. En nuestro ejemplo, `C:\Users\Alberto\Documents`.

2. Ejecutamos la siguiente sentencia SQL: `UPDATE ‘gestures‘.‘gesturemodels‘ SET ‘file‘ = REPLACE(‘file‘, ’C:\Program Files\Evaluador de Gestos’, ’C:\Users\Alberto\Documents\Evaluador de Gestos’);`

   Esta sentencia SQL cambia la ruta antigua de los gestos por la nueva ruta.

Apéndice D

Dependencias

D.1. Dependencias para la ejecución

El Evaluador de gestos y el Teleoperador de Robot Mindstorm necesitan un servidor mysql instalado en el equipo, con una base de datos llamada *gestures*, y un usuario *grs* con contraseña *gestures*.

D.2. Dependencias para el desarrollo

Si queremos continuar con el desarrollo de las soluciones del Evaluador de gestos o el Teleoperador de Robot Mindstorm, se necesitan varias librerías y kit de desarrollo. Es recomendable utilizar Visual Studio 2012 o una versión posterior e importar la solución.

Las siguientes librerías se pueden instalar en Visual Studio con el gestor de paquetes NuGet Packet Manager:

- EntityFramework 5
- Helix 3D Toolkit
- MySql.Data 6.7.4
- TCD.Kinect

- WPF Animated GIF
- MoreLINQ

Además, se necesitan:

- Kit de desarrollo de Kinect (Versión 1.8 o posterior)
- Plataform de Reconocimiento de Audio de Microsoft - Microsoft Speech Platform SDK (Versión 11 o posterior)

Apéndice E

Referencia de Clases por Proyecto

En este apéndice se encuentran todas las clases del proyecto con sus correspondientes métodos y la referencia a la página de la memoria donde se explica dicha clase.

E.1. Algorithms

- GestureRecognizer - Pag. 92

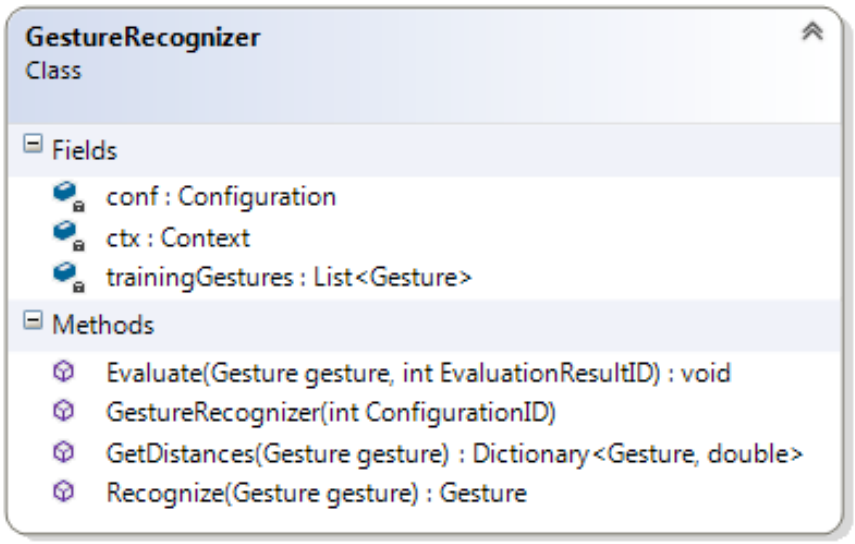


Figura E.1: GestureRecognizer

- Gesture - Pag. 78

Gesture
Class

Fields
- ctx : Context
- energies : List<double>
- lastSkeleton : Skeleton
- movementDetectionDistance : Distances
- movementDetectionPreprocesor : Preprocessor
- recorder : KinectRecorder
- replay : KinectReplay
- state : State

Properties
- energy { get; set; } : double
- ET_MovingToStopMoving { get; set; } : double
- ET_RestingToStartMoving { get; set; } : double
- ET_StartMovingToMoving { get; set; } : double
- ET_StartMovingToResting { get; set; } : double
- ET_StopMovingToMoving { get; set; } : double
- ET_StopMovingToResting { get; set; } : double
- gesturename { get; set; } : string
- ID { get; set; } : int
- Length { get; } : int
- Skeletons { get; set; } : List<Skeleton>
- TC_MovingToStopMoving { get; set; } : int
- TC_RestingToStartMoving { get; set; } : int
- TC_StartMovingToMoving { get; set; } : int
- TC_StartMovingToResting { get; set; } : int
- TC_StopMovingToMoving { get; set; } : int
- TC_StopMovingToResting { get; set; } : int
- TT_MovingToStopMoving { get; set; } : int
- TT_RestingToStartMoving { get; set; } : int
- TT_StartMovingToMoving { get; set; } : int
- TT_StartMovingToResting { get; set; } : int
- TT_StopMovingToMoving { get; set; } : int
- TT_StopMovingToResting { get; set; } : int
- username { get; set; } : string

Methods
- Delete() : void
- DeleteSkeletonFile() : void
- EvaluationRecord() : void
- Gesture(GestureModel gesture)
- Gesture(string gname, string uname, GestureType type)
- Gesture(string user, GestureType type)
- InitializeConf() : void
- Record(GestureType gesturetype) : void
- Record(SkeletonFrame frame) : void
- Replay(EventHandler<ReplaySkeletonFrameReadyEventArgs> eventHandler) : void
- Replay(EventHandler<ReplaySkeletonFrameReadyEventArgs> eventHandler, double speed) : void
- Replay(ReplaySkeletonFrame frame) : void
- ResetCounters() : void
- TrainingRecord() : void

Figura E.2: Gesture

E.1.1. Audio (*Audio*)

- SpeechRecognizer - Pag. 240

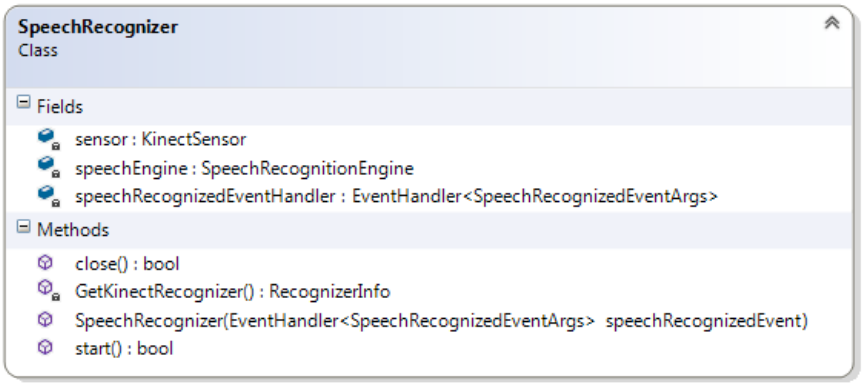


Figura E.3: SpeechRecognizer

E.1.2. DTW

- Distances - Pag. 76

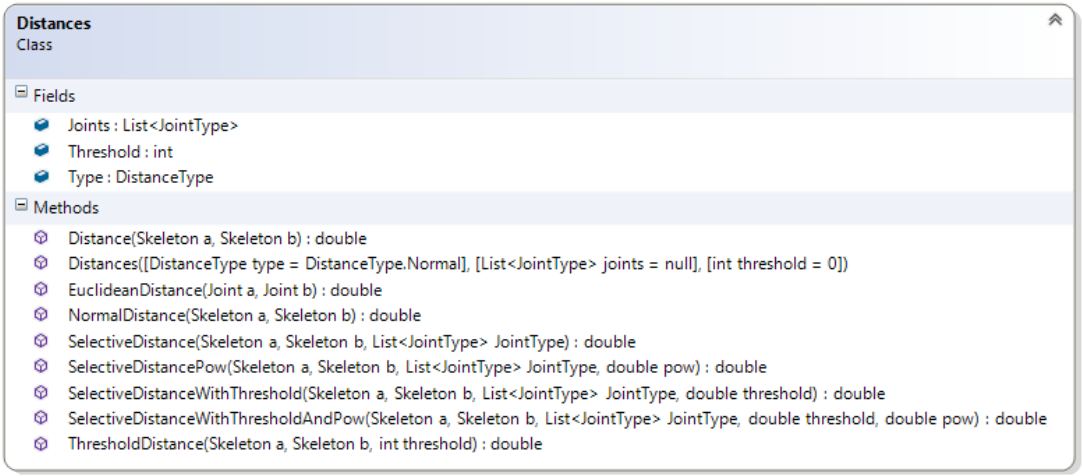


Figura E.4: Distances

- Dtw - Pag. 74

Dtw
Class

Fields
- _predecessorStepX : int[][][]
- _predecessorStepY : int[][][]
- boundaryConstraintEnd : bool
- boundaryConstraintStart : bool
- calculated : bool
- distance : Distances
- distances : double[][]
- isXLongerOrEqualThanY : bool
- pathCost : double[][]
- sakoeChibaConstraint : bool
- sakoeChibaMaxShift : int
- signalsLengthDifference : int
- slopeMatrixLookbehind : int
- slopeStepSizeAside : int
- slopeStepSizeDiagonal : int
- useSlopeConstraint : bool
- x : Gesture
- y : Gesture

Methods
- Calculate() : void
- CalculateDistances() : void
- CalculateWithoutSlopeConstraint() : void
- CalculateWithSlopeLimit() : void
- Dtw(Gesture _x, Gesture _y, ConfigurationDTW dtwConf)
- Dtw(Gesture _x, Gesture _y, DistanceType _distanceType, [bool _boundaryConstraintStart = true], [bool _boundaryConstraintEnd = true], [int? _slopeStepSizeDiagonal = null], [int? _slopeStepSizeAside = null], [int? _sakoeChibaMaxShift = null])
- GetCost() : double
- GetCostMatrix() : double[][]
- GetDistanceMatrix() : double[][]
- GetPath() : Tuple<int, int>[]
- InitializeArrays() : void

Figura E.5: Dtw

E.1.3. Extensiones *Extensions*

- SkeletonExtension - Pag. 93

SkeletonExtension
Class

Methods
- Center(Skeleton skeleton, SkeletonPoint center) : Skeleton
- CopySkeleton(Skeleton source, Skeleton destination) : void
- GetShortestRotationBetweenVectors(Vector3D vector1, Vector3D vector2) : Quaternion
- SkeletonPointToVector3D(SkeletonPoint pt) : Vector3D
- Translate(SkeletonPoint point, Vector3D translation) : SkeletonPoint
- Vector3DToSkeletonPoint(Vector3D pt) : SkeletonPoint
- VectorBetween(Skeleton skeleton, JointType startJoint, JointType endJoint) : Vector3D

Figura E.6: SkeletonExtension

- State - Pag. 93

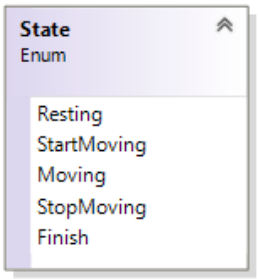


Figura E.7: State

E.1.4. Preprocesado (*Preprocessing*)

- Base - Pag. 91

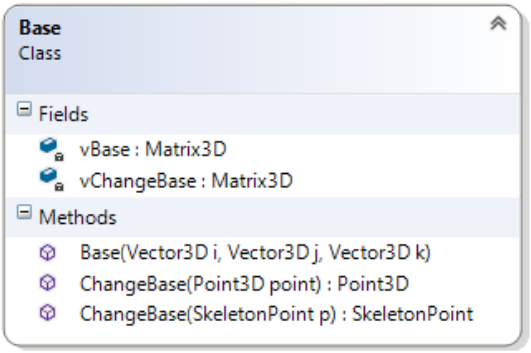


Figura E.8: Base

- CenterPreprocessor - Pag. 88

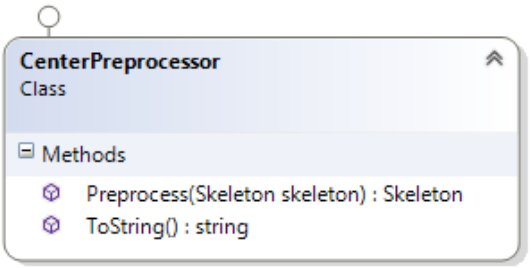


Figura E.9: CenterPreprocessor

- ChangeBasePreprocessor - Pag. 91

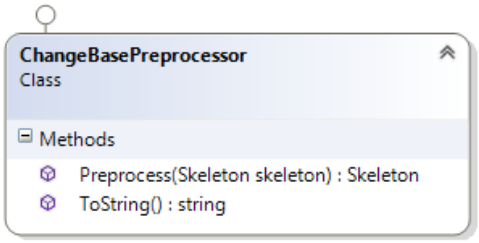


Figura E.10: ChangeBasePreprocessor

- IPreprocessor - Pag. 87

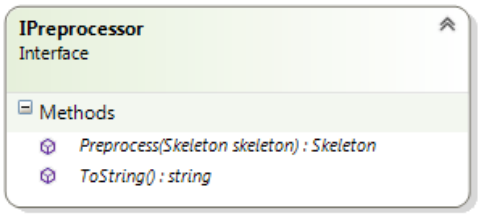


Figura E.11: IPreprocessor

- NonePreprocessor - Pag. 87

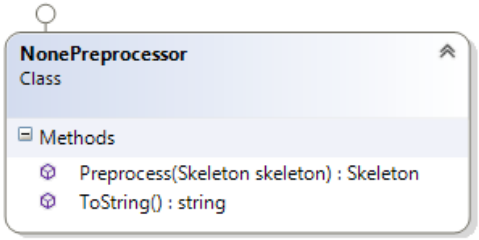


Figura E.12: NonePreprocessor

- RotatePreprocessor - Pag. 90

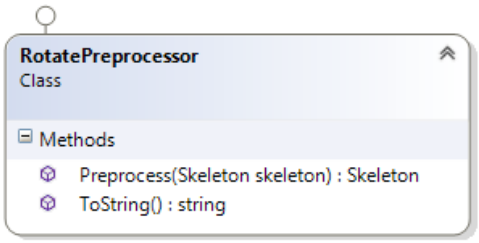


Figura E.13: RotatePreprocessor

- Preprocessor - Pag. 87

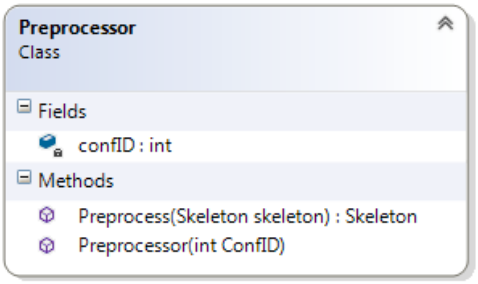


Figura E.14: Preprocessor

- ScalePreprocessor - Pag. 89

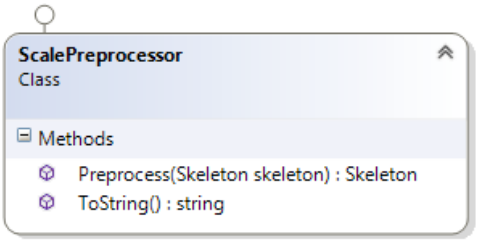


Figura E.15: ScalePreprocessor

E.1.5. Grabación (*Record*)

- KinectRecorder - Pag. 82

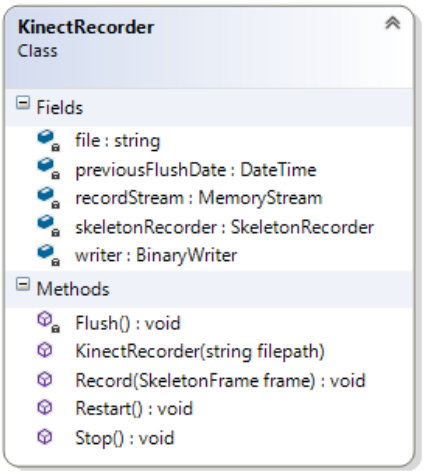


Figura E.16: KinectRecorder

- KinectReplay - Pag. 84

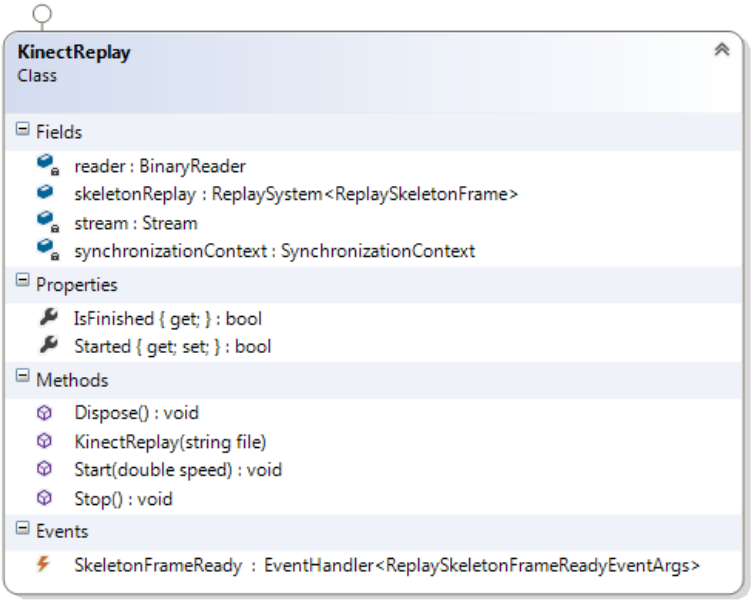


Figura E.17: KinectReplay

- ReplayFrame - Pag. 84

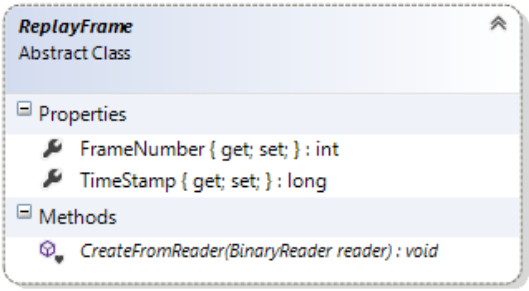


Figura E.18: ReplayFrame

- ReplaySkeletonFrameReadyEventArgs - Pag. 85

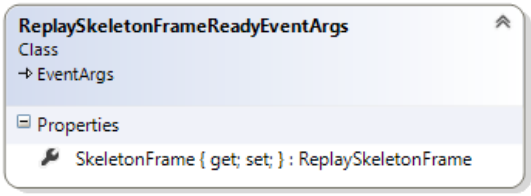


Figura E.19: ReplaySkeletonFrameReadyEventArgs

- ReplaySkeletonFrame - Pag. 84

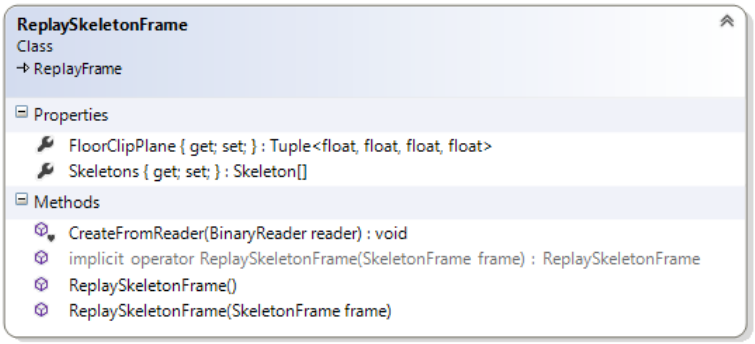


Figura E.20: ReplaySkeletonFrame

- ReplaySystem - Pag. 84

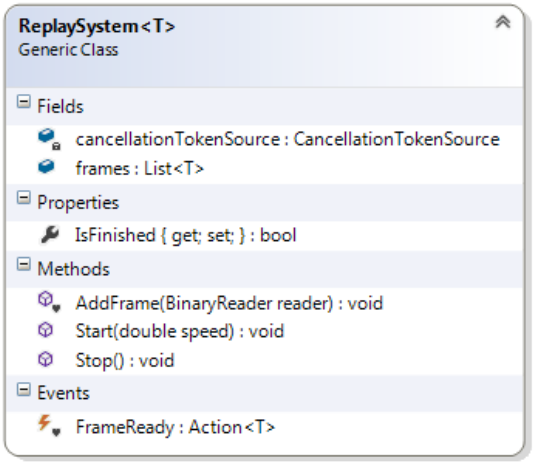


Figura E.21: ReplaySystem

- SkeletonRecorder - Pag. 82

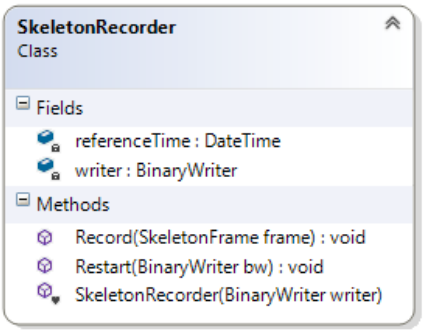


Figura E.22: SkeletonRecorder

E.2. GestureEvaluator

E.2.1. Ventanas

- HomeWindow - Pag. 98

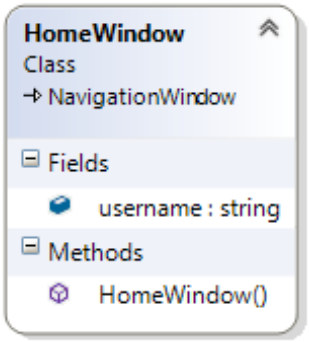


Figura E.23: HomeWindow

- MainWindow - Pag. 95

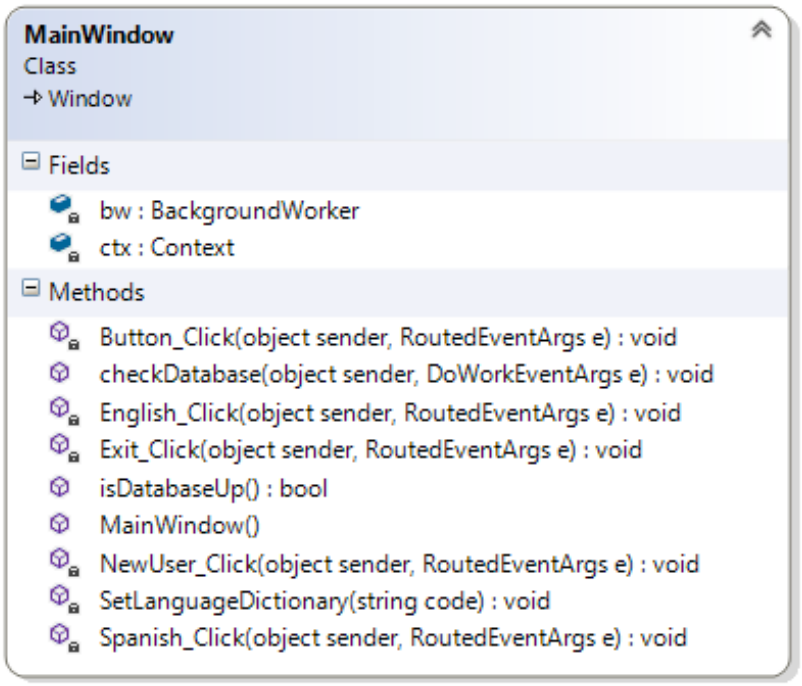


Figura E.24: MainWindow

- RegisterWindow - Pag. 96

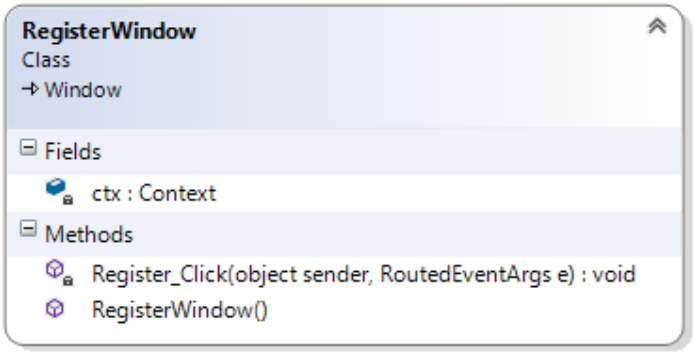


Figura E.25: RegisterWindow

E.2.2. Páginas (*Pages*)

- HomeWindow - Pag. 98

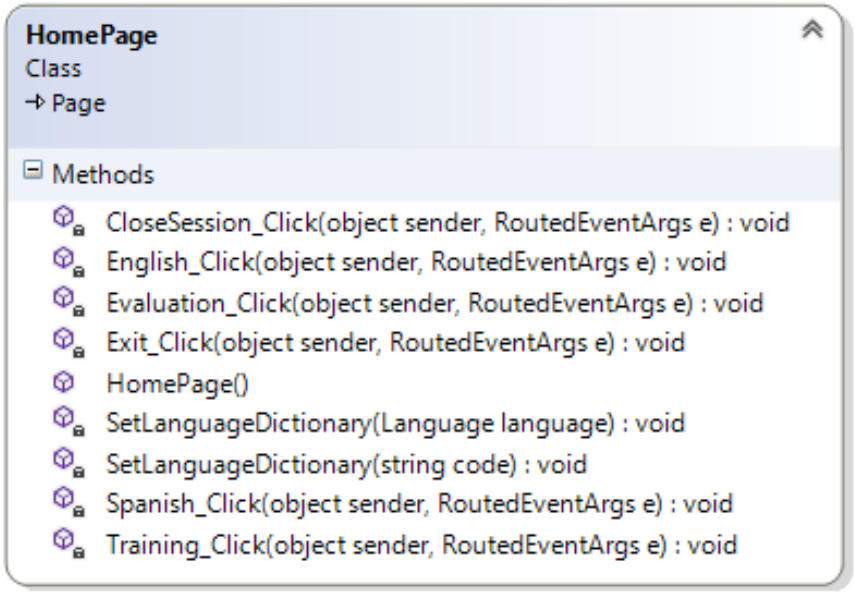


Figura E.26: HomePage

Detección (*Detection*)

Configuración (*Config*)

- CustomEvaluationConfigurationPage - Pag. 127

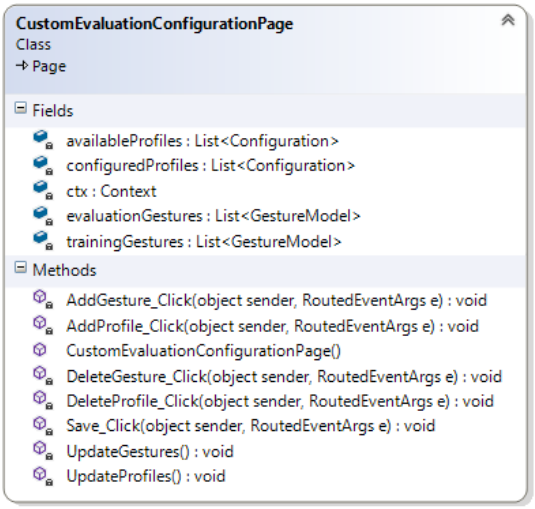


Figura E.27: CustomEvaluationConfigurationPage

- EvaluationConfigurationPage - Pag. 124

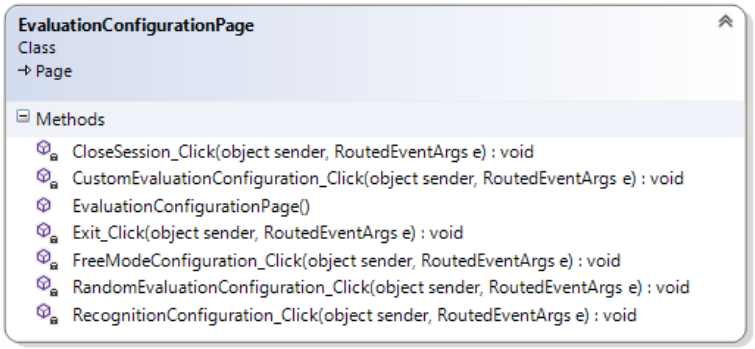


Figura E.28: EvaluationConfigurationPage

- FreeModeConfigurationPage - Pag. 125

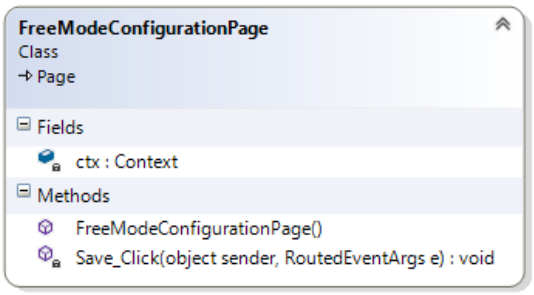


Figura E.29: FreeModeConfigurationPage

- RandomEvaluationConfigurationPage - Pag. 126

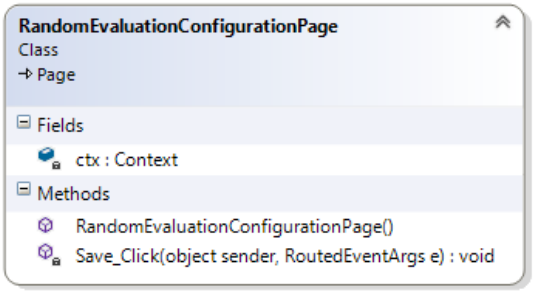


Figura E.30: RandomEvaluationConfigurationPage

Configuración de Reconocimiento de Gestos (*Recognition*)

- DTWConfigurationPage - Pag. 132

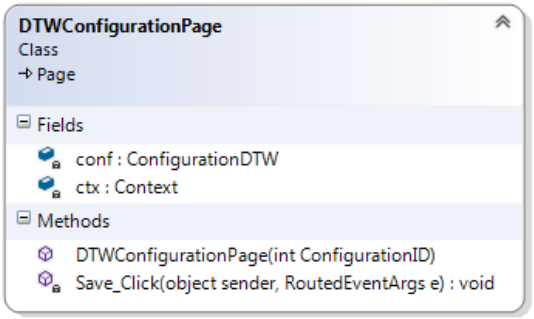


Figura E.31: DTWConfigurationPage

- MovementConfigurationPage - Pag. 133

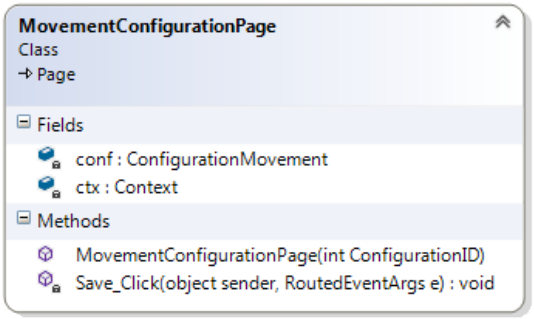


Figura E.32: MovementConfigurationPage

- PreprocessConfigurationPage - Pag. 134

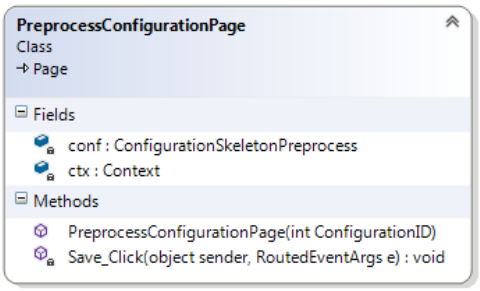


Figura E.33: PreprocessConfigurationPage

- ProfileConfigurationPage - Pag. 131

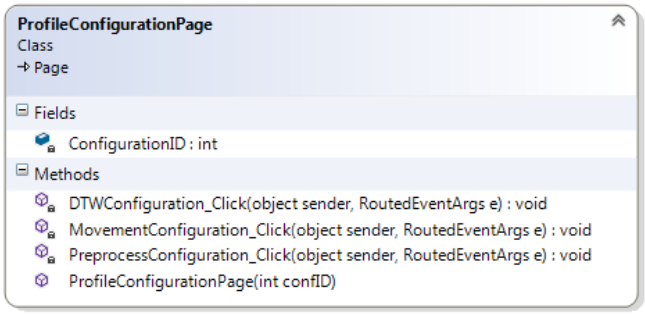


Figura E.34: ProfileConfigurationPage

- RecognitionConfigurationPage - Pag. 129

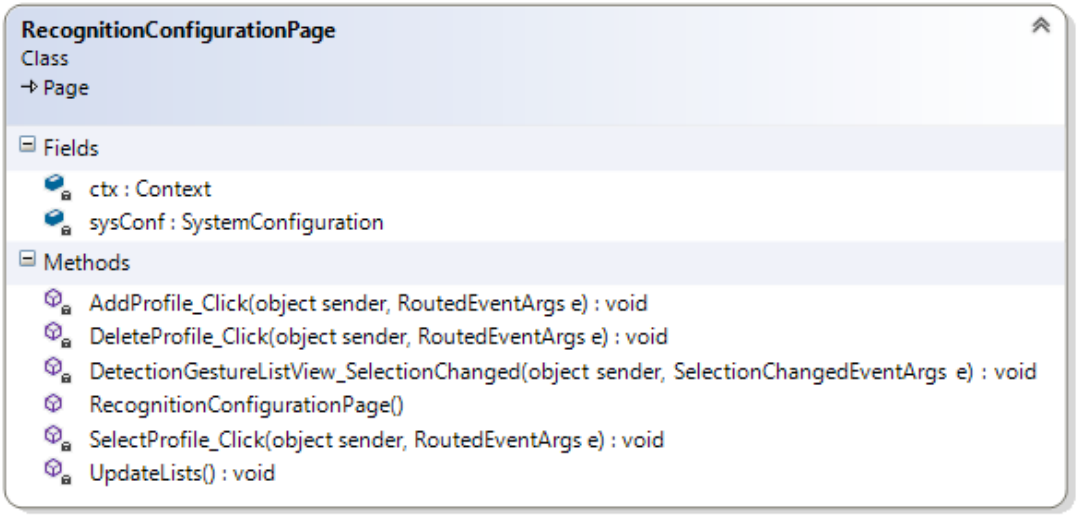


Figura E.35: RecognitionConfigurationPage

Evaluación (*EvaluationMode*)

- CalculateEvaluationResultsPage - Pag. 119

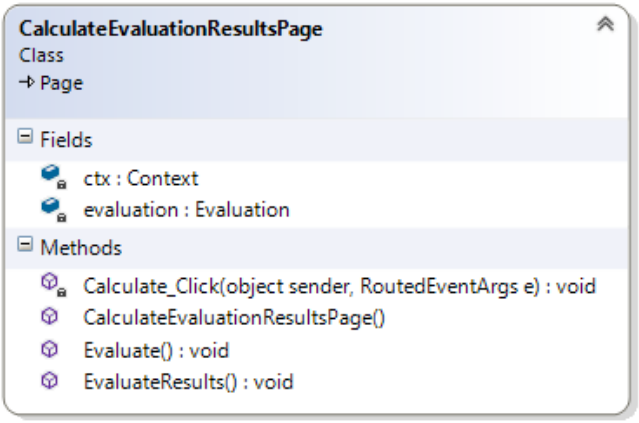


Figura E.36: CalculateEvaluationResultsPage

- EvaluationHomePage - Pag. 115

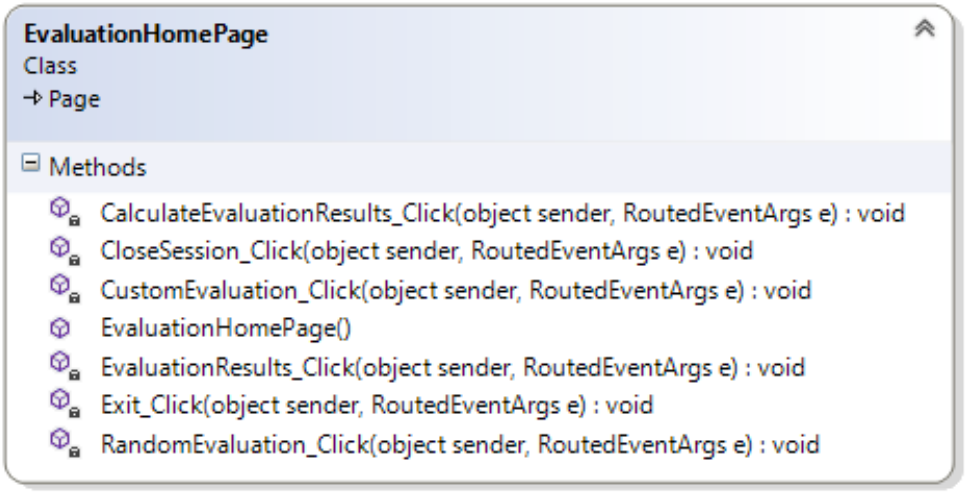


Figura E.37: EvaluationHomePage

- EvaluationPage - Pag. 116

EvaluationPage
Class
→ Page

Fields

- automatic : bool
- counter : int
- ctx : Context
- currentGesture : Gesture
- currentTrainingGesture : GestureModel
- enableRecord : bool
- evaluation : Evaluation
- images : Dictionary<int, string>
- numGestures : int
- sensor : KinectSensor
- trainingGestures : IEnumerator<KeyValuePair<int, string>>

Methods

- ButtonFinish_Click(object sender, RoutedEventArgs e) : void
- EvaluationPage(EvaluationType evaluationType)
- finishGesture() : void
- LoadGif() : void
- NavigationService_Navigating(object sender, NavigatingCancelEventArgs e) : void
- Next_Click(object sender, RoutedEventArgs e) : void
- NextGesture() : void
- Replay_Click(object sender, RoutedEventArgs e) : void
- ReplayGif_Click(object sender, RoutedEventArgs e) : void
- ReRecord_Click(object sender, RoutedEventArgs e) : void
- sensor_SkeletonFrameReadyRecord(object sender, SkeletonFrameReadyEventArgs e) : void
- sensor_SkeletonFrameReadyReplay(object sender, ReplaySkeletonFrameReadyEventArgs e) : void

Figura E.38: EvaluationPage

- EvaluationResultPage - Pag. 120

EvaluationResultPage
Class
→ Page

Fields

- ctx : Context
- evaluation : Evaluation
- evaluationItemResults : List<EvaluationItemResult>
- evaluationResults : List<EvaluationResult>
- evaluationResultsDTW : List<EvaluationItemResultDTW>
- gestureDetected : Gesture
- gestureRecorded : Gesture
- gestureReplayed : Gesture
- preprocessor : Preprocessor
- user : User

Methods

- DTWResultsList_SelectionChanged(object sender, SelectionChangedEventArgs e) : void
- EvaluationListView_SelectionChanged(object sender, SelectionChangedEventArgs e) : void
- EvaluationResultPage([int evaluationID = -1])
- EvaluationResultsList_SelectionChanged(object sender, SelectionChangedEventArgs e) : void
- GestureDetectedCombobox_SelectionChanged(object sender, SelectionChangedEventArgs e) : void
- GestureDetectedPlay_Click(object sender, RoutedEventArgs e) : void
- GestureRecordedCombobox_SelectionChanged(object sender, SelectionChangedEventArgs e) : void
- GestureRecordedPlay_Click(object sender, RoutedEventArgs e) : void
- GestureReplayedCombobox_SelectionChanged(object sender, SelectionChangedEventArgs e) : void
- GestureReplayedPlay_Click(object sender, RoutedEventArgs e) : void
- Initialize(int evaluationID) : void
- PlayGestureDetected() : void
- PlayGestureRecorded() : void
- PlayGestureReplayed() : void
- PlayGestures() : void
- PreprocessGestureReplayed() : void
- ProfileList_SelectionChanged(object sender, SelectionChangedEventArgs e) : void
- sensor_SkeletonFrameReadyGestureDetected(object sender, ReplaySkeletonFrameReadyEventArgs e) : void
- sensor_SkeletonFrameReadyGestureRecorded(object sender, ReplaySkeletonFrameReadyEventArgs e) : void
- sensor_SkeletonFrameReadyGestureReplayed(object sender, ReplaySkeletonFrameReadyEventArgs e) : void

Figura E.39: EvaluationResultPage

Detección continua (*FreeMode*)

- DetectionPage - Pag. 112

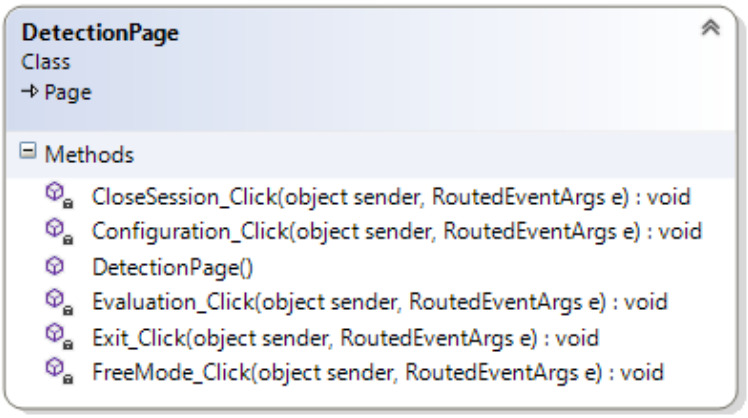


Figura E.40: DetectionPage

- FreeModePage - Pag. 112

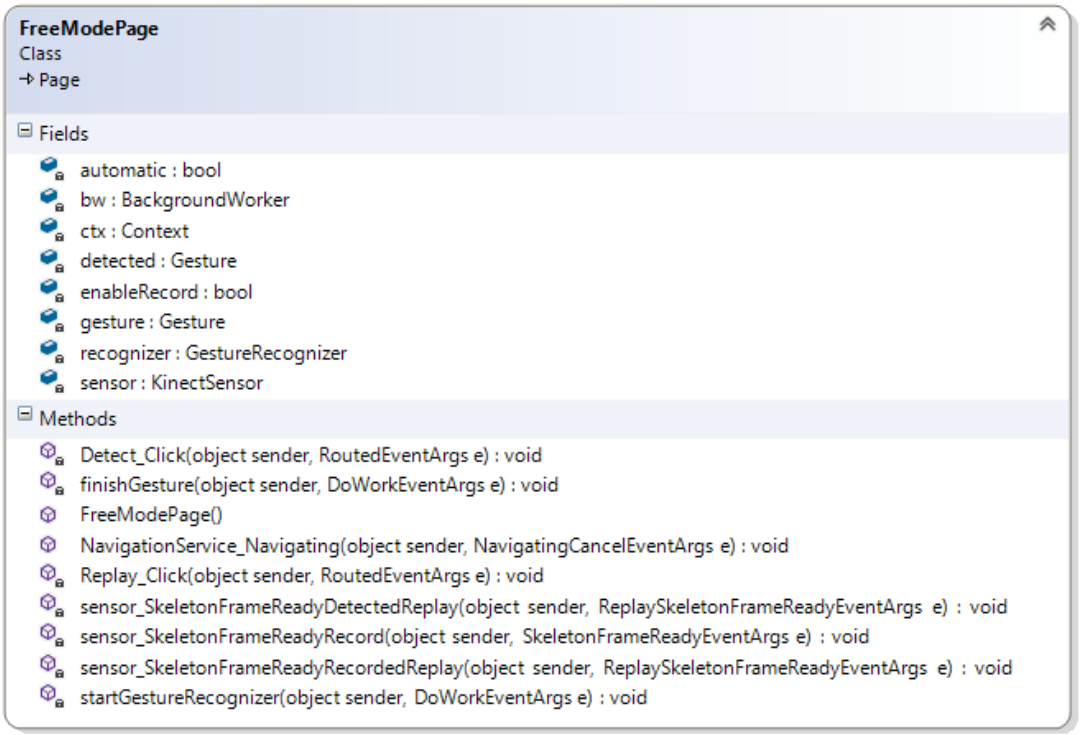


Figura E.41: FreeModePage

Entrenamiento (*Training*)

- StartTrainingPage - Pag. 104

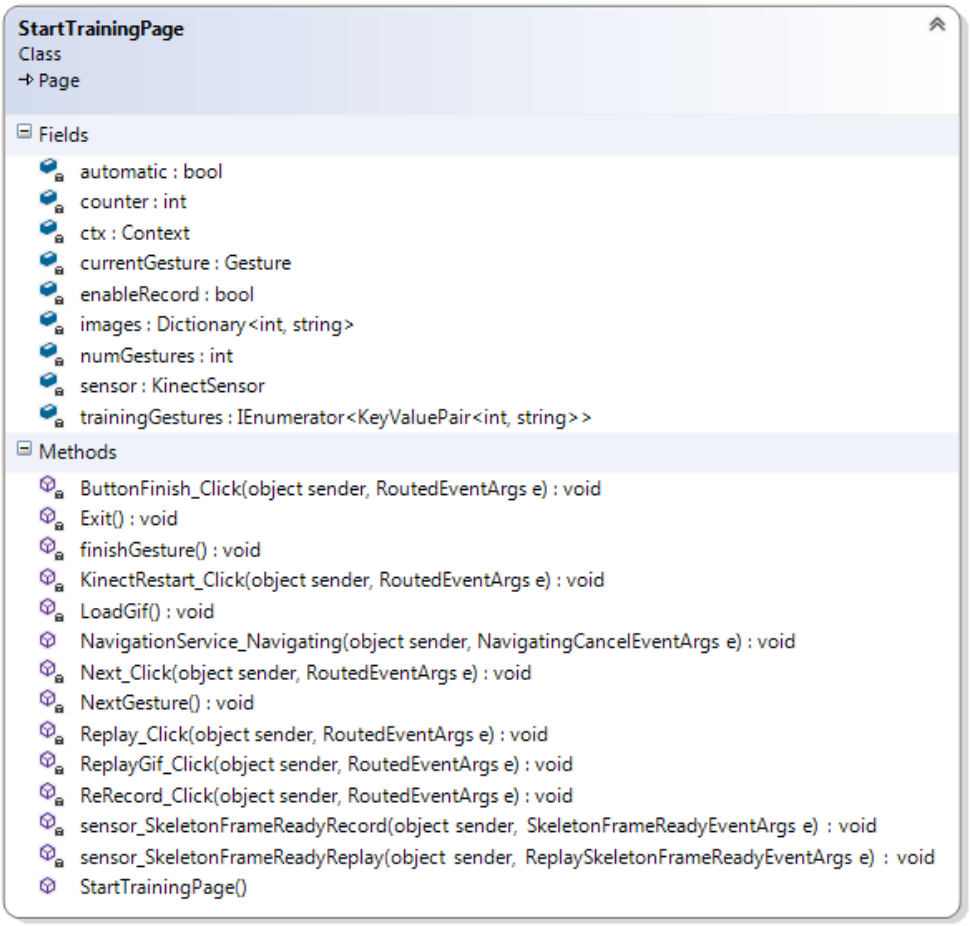


Figura E.42: StartTrainingPage

- TrainingHomePage - Pag. 100

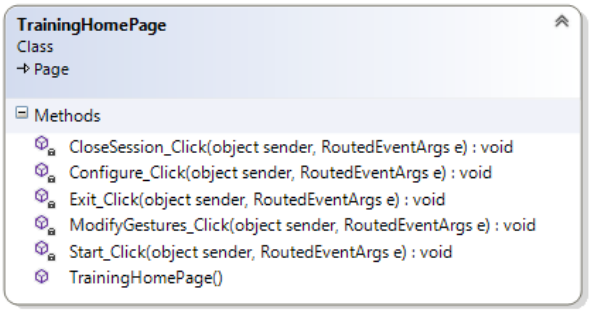


Figura E.43: TrainingHomePage

Configuración (*Configuration*)

- ConfigurationPage - Pag. 101

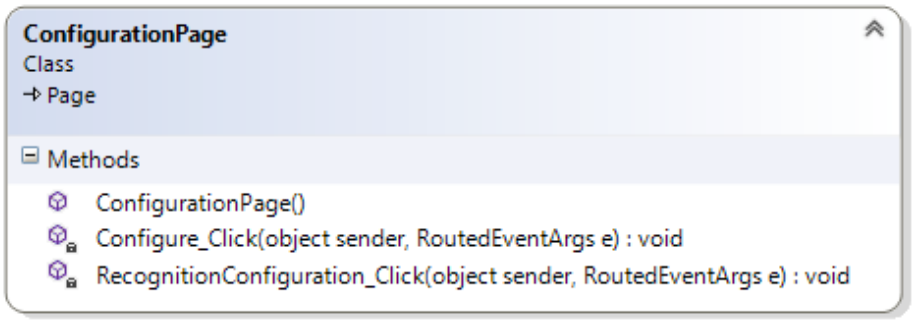


Figura E.44: ConfigurationPage

- ConfigurationTrainingPage - Pag. 102

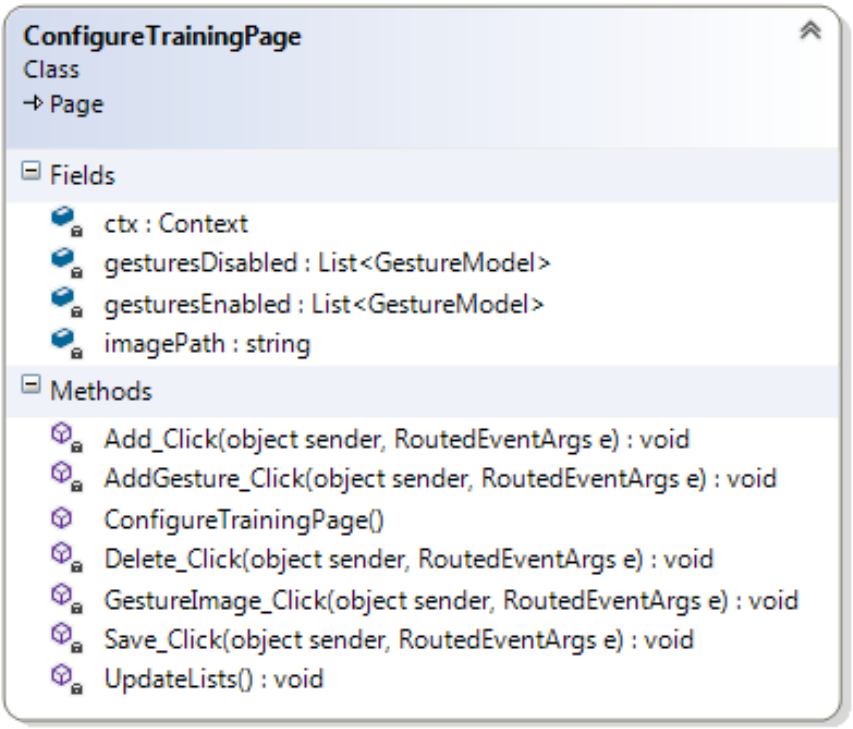


Figura E.45: ConfigurationTrainingPage

Gestos (*Gestures*)

- AddGesturesPage - Pag. 108

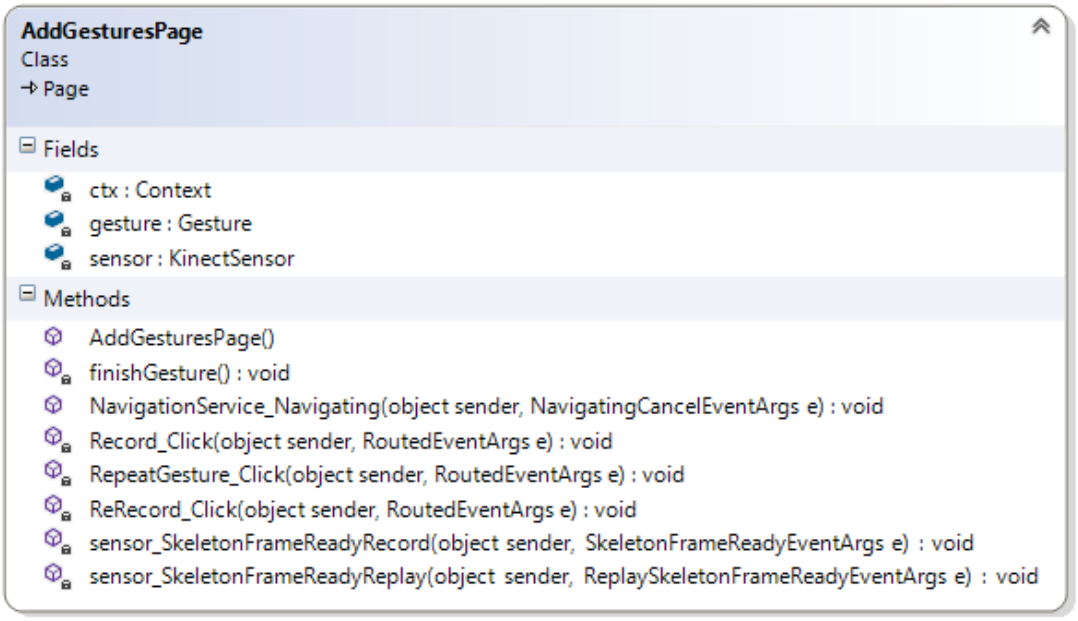


Figura E.46: AddGesturesPage

- DeleteGesturesPage - Pag. 109

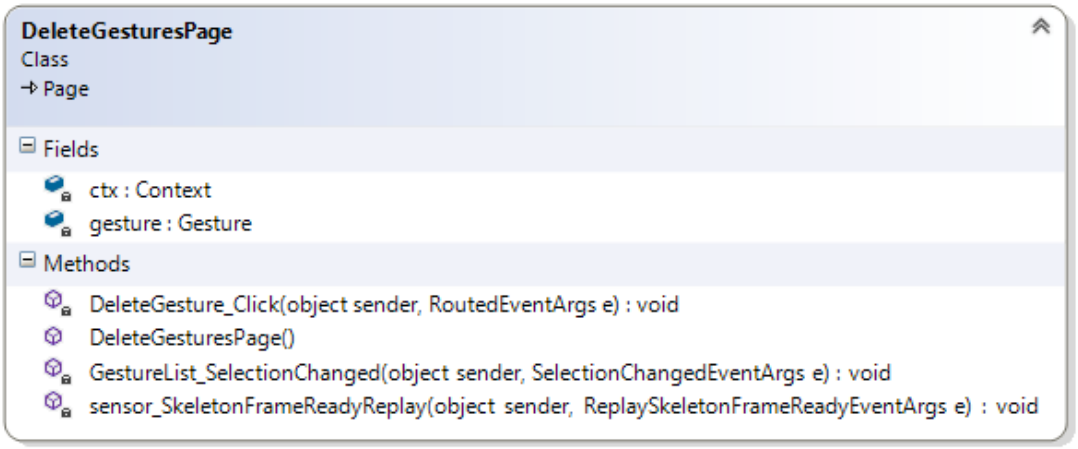


Figura E.47: DeleteGesturesPage

- GesturesPage - Pag. 107

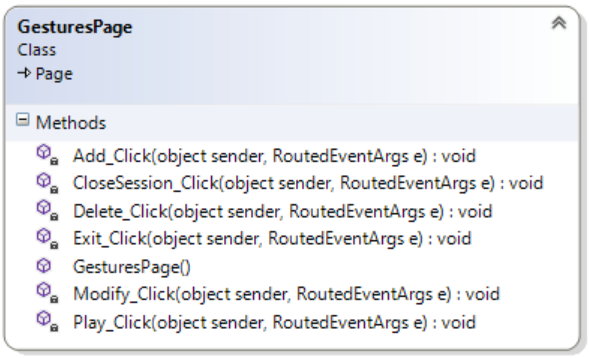


Figura E.48: GesturesPage

- ModifyGesturesPage - Pag. 110

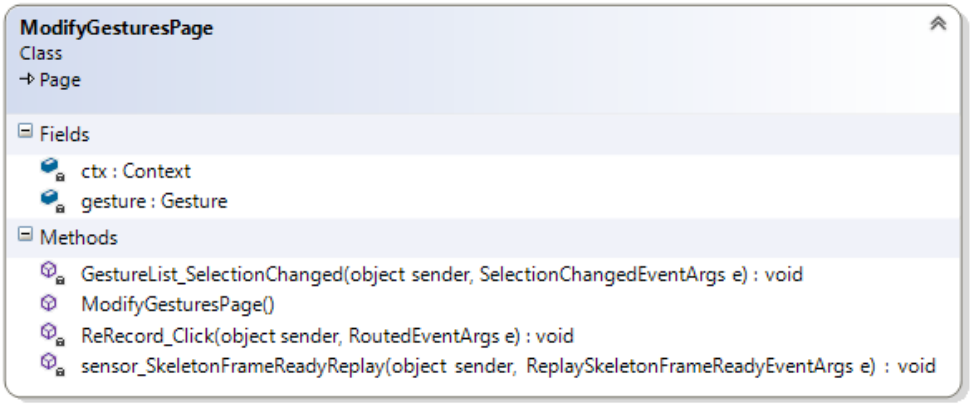


Figura E.49: ModifyGesturesPage

- PlayGesturePage - Pag. 111

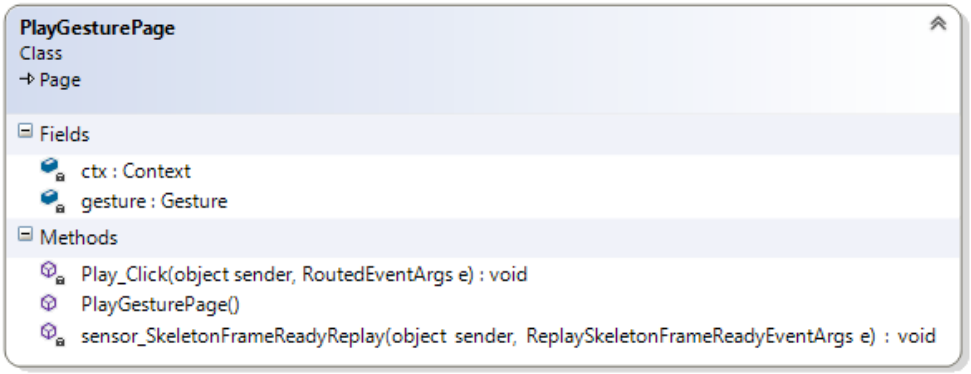


Figura E.50: PlayGesturePage

E.2.3. Extensiones (*Helpers*)

- RedToGreenScaleConverter - Pag. 251

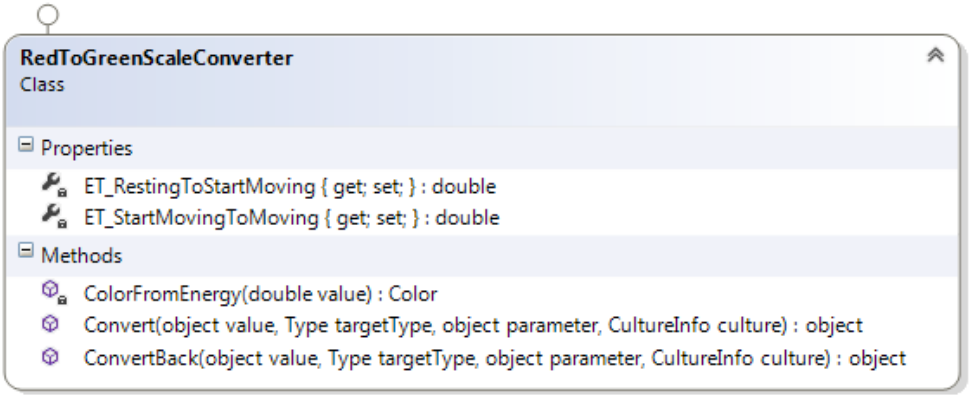


Figura E.51: RedToGreenScaleConverter

E.3. MindstormController

E.3.1. Ventanas

- HomeWindow - Pag. 245

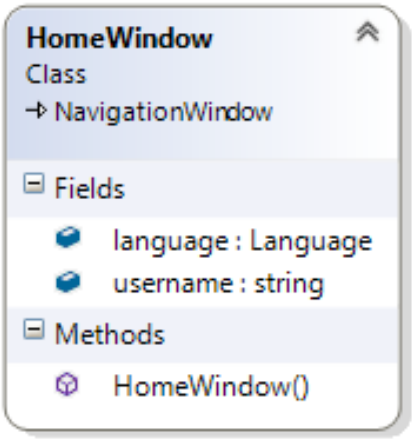


Figura E.52: HomeWindow

- MainWindow - Pag. 242

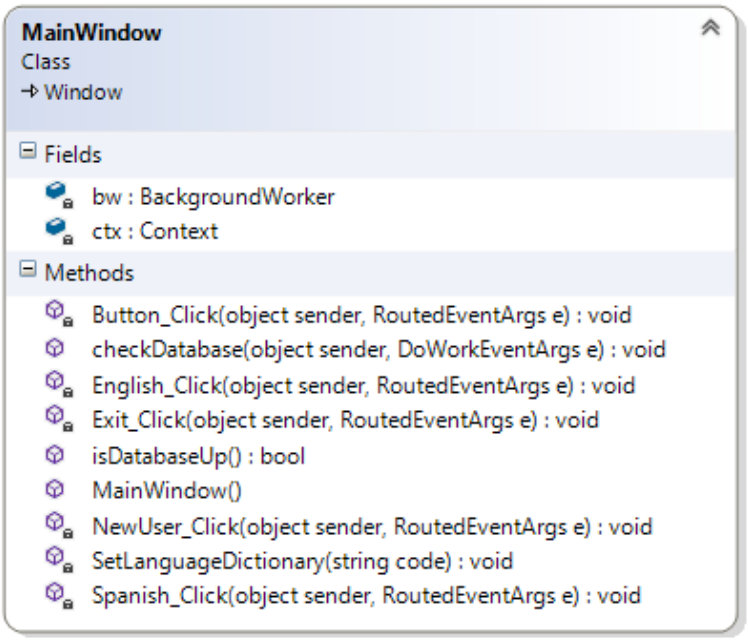


Figura E.53: MainWindow

- RegisterWindow - Pag. 243

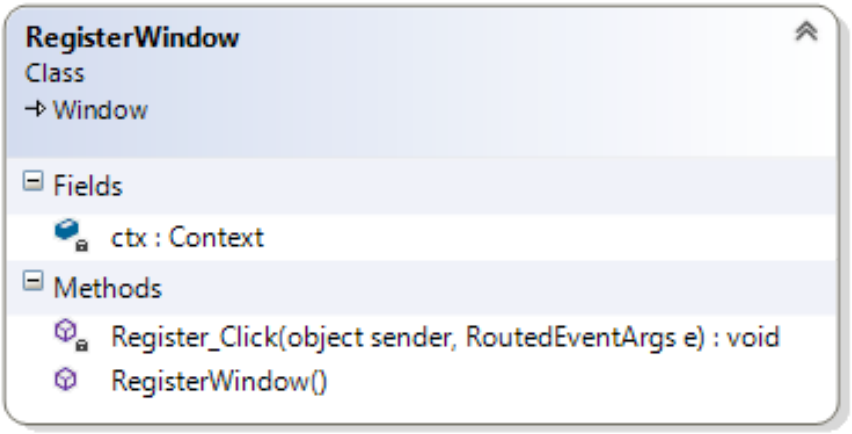


Figura E.54: RegisterWindow

E.3.2. Páginas (*Pages*)

- ConfigurationPage - Pag. 247

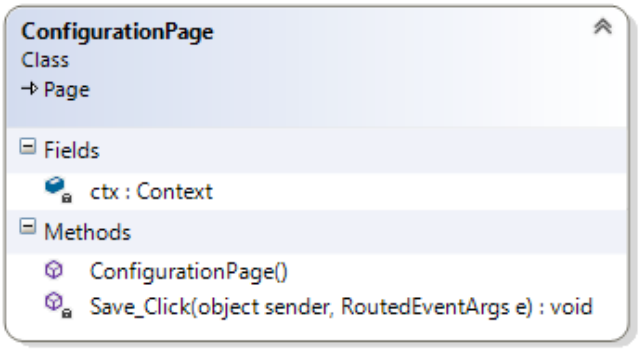


Figura E.55: ConfigurationPage

- HomePage - Pag. 245

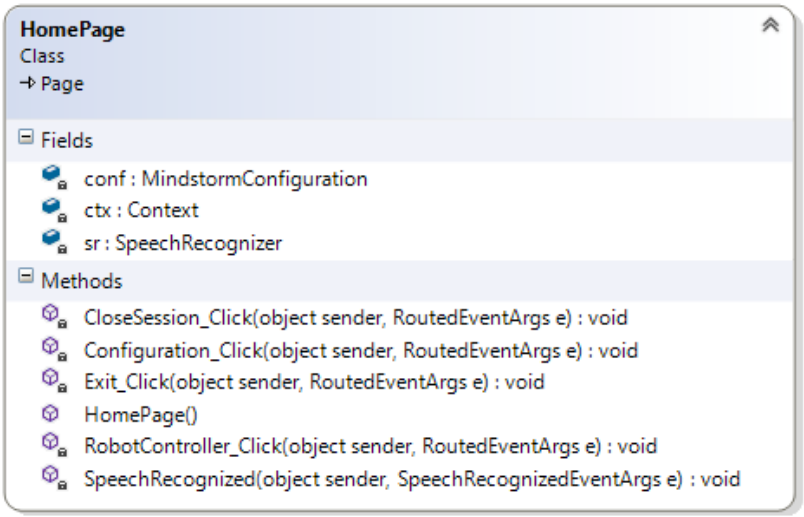


Figura E.56: HomePage

- RobotControllerPage - Pag. 249

RobotControllerPage
Class
→ Page

Fields

- bw1 : BackgroundWorker
- bw2 : BackgroundWorker
- conf : MindstormConfiguration
- ctx : Context
- enableRecord : bool
- ev3 : Brick<Sensor, Sensor, Sensor, Sensor>
- gesture : Gesture
- gestureRecognizer : GestureRecognizer
- leftArm : MotorState
- leftFoot : MotorState
- rightArm : MotorState
- rightFoot : MotorState
- sensor : KinectSensor
- speechRecognizer : SpeechRecognizer

Methods

- BothArmsDown() : void
- BothArmsDownCommand(object sender, DoWorkEventArgs e) : void
- BothArmsMove() : void
- BothArmsMoveCommand(object sender, DoWorkEventArgs e) : void
- BothArmsUp() : void
- BothArmsUpCommand(object sender, DoWorkEventArgs e) : void
- BothFeetBackward() : void
- BothFeetBackwardCommand(object sender, DoWorkEventArgs e) : void
- BothFeetForward() : void
- BothFeetForwardCommand(object sender, DoWorkEventArgs e) : void
- BothFeetMove() : void
- BothFeetMoveCommand(object sender, DoWorkEventArgs e) : void
- finishGesture(object sender, DoWorkEventArgs e) : void
- Move(string name, string motor, sbyte speed) : void
- MoveDual(string name, string motor, sbyte speed) : void
- NavigationService_Navigating(object sender, NavigatingCancelEventArgs e) : void
- RobotControllerPage()
- SendCommand(object sender, DoWorkEventArgs e) : void
- SendMoveCommand(object sender, DoWorkEventArgs e) : void
- sensor_SkeletonFrameReadyRecord(object sender, SkeletonFrameReadyEventArgs e) : void
- SpeechRecognized(object sender, SpeechRecognizedEventArgs e) : void
- startGestureRecognizer(object sender, DoWorkEventArgs e) : void
- WaitForMotorToStop(Brick<Sensor, Sensor, Sensor, Sensor> ev3, string motor) : void
- WriteText(string text) : void

Figura E.57: RobotControllerPage

E.3.3. Extensiones (*Helpers*)

- RedToGreenScaleConverter - Pag. 251

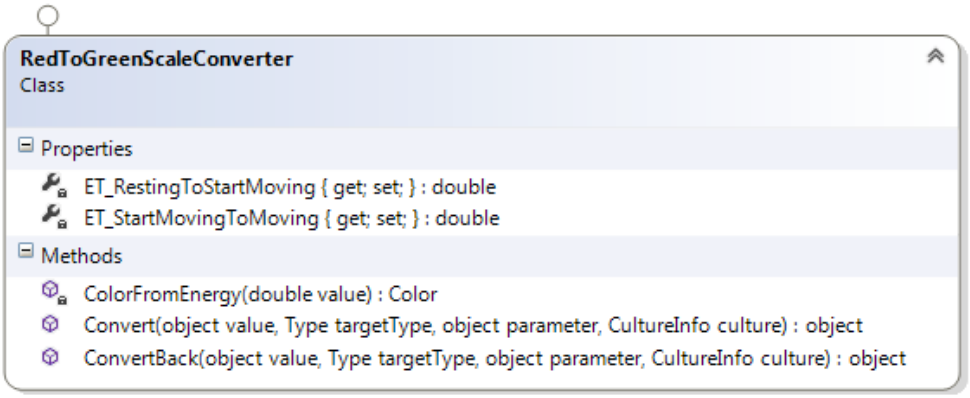


Figura E.58: RedToGreenScaleConverter

- MotorState - Pag. 251

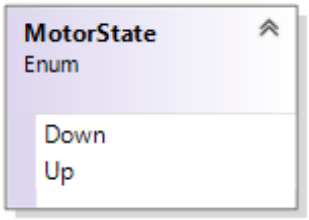


Figura E.59: MotorState

E.4. Models

- Initializer - Pag. 139
- Context - Pag. 139

E.4.1. Tablas Principales

- GestureModel - Pag. 140

Add
Remove
Clear
Up
Down

Name	Type / Length	Algorithm
PRIMARY KEY	PRIMARY	
ID	UNIQUE	
user_ID	KEY	

Columns: Add Remove Up Down

#	Name	Datatype	Length/Set	Unsign...	Allow NULL	Zerofill	Default	Comment	Collation
1	ID	INT	11	☐	☐	☐	AUTO_INCREMENT		
2	type	INT	11	☐	☐	☐	No default		
3	name	LONGTEXT			☑		No default		
4	image	LONGTEXT			☑		No default		
5	file	LONGTEXT			☑		No default		
6	user_ID	INT	11	☐	☑	☐	NULL		

Figura E.60: Esquema de la tabla Gesturemodel

- User - Pag. 140

Add
Remove
Clear
Up
Down

Name	Type / Length	Algorithm
PRIMARY KEY	PRIMARY	
ID	UNIQUE	

Columns: Add Remove Up Down

#	Name	Datatype	Length/Set	Unsign...	Allow NULL	Zerofill	Default	Comment	Collation
1	ID	INT	11	☐	☐	☐	AUTO_INCREMENT		
2	name	LONGTEXT			☑		No default		
3	password	LONGTEXT			☑		No default		
4	language	INT	11	☐	☐	☐	No default		

Figura E.61: Esquema de la tabla User

E.4.2. Tablas de Configuraciones

- Configuration - Pag. 141

Add / Remove / Clear / Up / Down

Name	Type / Length	Algorithm
PRIMARY KEY	PRIMARY	
ID	UNIQUE	
user_ID	KEY	
gestureRecognizerConfiguration_ID	KEY	
dtwConfiguration_ID	KEY	
movementConfiguration_ID	KEY	
skeletonPreprocessConfiguration_ID	KEY	

Columns: Add Remove Up Down

#	Name	Datatype	Length/Set	Unsign...	Allow NULL	Zerofill	Default	Comment	Collation
1	ID	INT	11	☐	☐	☐	AUTO_INCREMENT		
2	name	LONGTEXT			☑		No default		
3	type	INT	11	☐	☐	☐	No default		
4	user_ID	INT	11	☐	☑	☐	NULL		
5	gestureRecognizerConfiguration_ID	INT	11	☐	☑	☐	NULL		
6	dtwConfiguration_ID	INT	11	☐	☑	☐	NULL		
7	movementConfiguration_ID	INT	11	☐	☑	☐	NULL		
8	skeletonPreprocessConfiguration...	INT	11	☐	☑	☐	NULL		

Figura E.62: Esquema de la tabla Configuration

- ConfigurationDTW - Pag. 142

Add / Remove / Clear / Up / Down

Name	Type / Length	Algorithm
PRIMARY KEY	PRIMARY	
ID	UNIQUE	

Columns: Add Remove Up Down

#	Name	Datatype	Length/Set	Unsign...	Allow NULL	Zerofill	Default	Comment	Collation
1	ID	INT	11	☐	☐	☐	AUTO_INCREMENT		
2	distance	INT	11	☐	☐	☐	No default		
3	boundaryConstraintStart	TINYINT	1	☐	☐	☐	No default		
4	boundaryConstraintEnd	TINYINT	1	☐	☐	☐	No default		
5	slopeStepSizeDiagonal	INT	11	☐	☑	☐	NULL		
6	slopeStepSizeAside	INT	11	☐	☑	☐	NULL		
7	sakoeChibaMaxShift	INT	11	☐	☑	☐	NULL		

Figura E.63: Esquema de la tabla ConfigurationDTW

- ConfigurationGestureRecognizer - Pag. 143

Add / Remove / Clear / Up / Down

Name	Type / Length	Algorithm
PRIMARY KEY	PRIMARY	
ID	UNIQUE	

Columns: Add Remove Up Down

#	Name	Datatype	Length/Set	Unsign...	Allow NULL	Zerofill	Default	Comment	Collation
1	ID	INT	11	☐	☐	☐	AUTO_INCREMENT		
2	distanceRecognizerThreshold	DOUBLE		☐	☑	☐	NULL		

Figura E.64: Esquema de la tabla ConfigurationGestureRecognizer

- ConfigurationMovement - Pag. 143

Add / Remove / Clear / Up / Down

Name	Type / Length	Algorithm
PRIMARY KEY	PRIMARY	
ID	UNIQUE	

Columns: Add Remove Up Down

#	Name	Datatype	Length/Set	Unsign...	Allow NULL	Zerofill	Default	Comment	Collation
1	ID	INT	11				AUTO_INCREMENT		
2	ET_RestingToStartMoving	DOUBLE					No default		
3	ET_StartMovingToResting	DOUBLE					No default		
4	ET_StartMovingToMoving	DOUBLE					No default		
5	ET_MovingToStopMoving	DOUBLE					No default		
6	ET_StopMovingToMoving	DOUBLE					No default		
7	ET_StopMovingToResting	DOUBLE					No default		
8	TT_RestingToStartMoving	INT	11				No default		
9	TT_StartMovingToResting	INT	11				No default		
10	TT_StartMovingToMoving	INT	11				No default		
11	TT_MovingToStopMoving	INT	11				No default		
12	TT_StopMovingToMoving	INT	11				No default		
13	TT_StopMovingToResting	INT	11				No default		

Figura E.65: Esquema de la tabla ConfigurationMovement

- ConfigurationSkeletonPreprocess - Pag. 144

Add / Remove / Clear / Up / Down

Name	Type / Length	Algorithm
PRIMARY KEY	PRIMARY	
ID	UNIQUE	

Columns: Add Remove Up Down

#	Name	Datatype	Length/Set	Unsign...	Allow NULL	Zerofill	Default	Comment	Collation
1	ID	INT	11				AUTO_INCREMENT		
2	Center	TINYINT	1				No default		
3	Scale	TINYINT	1				No default		
4	Rotate	INT	11				No default		

Figura E.66: Esquema de la tabla ConfigurationSkeletonPreprocess

- SystemConfiguration - Pag. 145

Add / Remove / Clear / Up / Down

Name	Type / Length	Algorithm
PRIMARY KEY	PRIMARY	
ID	UNIQUE	
user_ID	KEY	
movementDetectionProfile_ID	KEY	
freeModeDetectionProfile_ID	KEY	

Columns: Add Remove Up Down

#	Name	Datatype	Length/Set	Unsign...	Allow NULL	Zerofill	Default	Comment	Collation
1	ID	INT	11				AUTO_INCREMENT		
2	trainingAutomatic	TINYINT	1				No default		
3	freeModeAutomatic	TINYINT	1				No default		
4	randomEvaluationNumOfGestures	INT	11				No default		
5	randomEvaluationAutomatic	TINYINT	1				No default		
6	randomEvaluationProfileSerialized	LONGTEXT			☑		No default		
7	customEvaluationAutomatic	TINYINT	1				No default		
8	customEvaluationGestureSerialized	LONGTEXT			☑		No default		
9	customEvaluationProfileSerialized	LONGTEXT			☑		No default		
10	user_ID	INT	11		☑		NULL		
11	movementDetectionProfile_ID	INT	11		☑		NULL		
12	freeModeDetectionProfile_ID	INT	11		☑		NULL		

Figura E.67: Esquema de la tabla SystemConfiguration

- MindstormConfiguration - Pag. 252

Add
Remove
Clear
Up
Down

Name	Type / Length	Algorithm
PRIMARY KEY	PRIMARY	
ID	UNIQUE	
user_ID	KEY	
profile_ID	KEY	
rightArmMove_ID	KEY	
rightArmUp_ID	KEY	
rightArmDown_ID	KEY	
leftArmMove_ID	KEY	
leftArmUp_ID	KEY	
leftArmDown_ID	KEY	
bothArmsMove_ID	KEY	
bothArmsUp_ID	KEY	
bothArmsDown_ID	KEY	
rightFootMove_ID	KEY	
rightFootForward_ID	KEY	
rightFootBackward_ID	KEY	
leftFootMove_ID	KEY	
leftFootForward_ID	KEY	
leftFootBackward_ID	KEY	
bothFeetMove_ID	KEY	
bothFeetForward_ID	KEY	
bothFeetBackward_ID	KEY	

Columns: Add Remove Up Down

#	Name	Datatype	Length/Set	Unsign...	Allow NULL	Zerofill	Default	Comment	Collation
1	ID	INT	11	☐	☐	☐	AUTO_INCREMENT		
2	connection	INT	11	☐	☐	☐	No default		
3	recognition	INT	11	☐	☐	☐	No default		
4	user_ID	INT	11	☐	☑	☐	NULL		
5	profile_ID	INT	11	☐	☑	☐	NULL		
6	rightArmMove_ID	INT	11	☐	☑	☐	NULL		
7	rightArmUp_ID	INT	11	☐	☑	☐	NULL		
8	rightArmDown_ID	INT	11	☐	☑	☐	NULL		
9	leftArmMove_ID	INT	11	☐	☑	☐	NULL		
10	leftArmUp_ID	INT	11	☐	☑	☐	NULL		
11	leftArmDown_ID	INT	11	☐	☑	☐	NULL		
12	bothArmsMove_ID	INT	11	☐	☑	☐	NULL		
13	bothArmsUp_ID	INT	11	☐	☑	☐	NULL		
14	bothArmsDown_ID	INT	11	☐	☑	☐	NULL		
15	rightFootMove_ID	INT	11	☐	☑	☐	NULL		
16	rightFootForward_ID	INT	11	☐	☑	☐	NULL		
17	rightFootBackward_ID	INT	11	☐	☑	☐	NULL		
18	leftFootMove_ID	INT	11	☐	☑	☐	NULL		
19	leftFootForward_ID	INT	11	☐	☑	☐	NULL		
20	leftFootBackward_ID	INT	11	☐	☑	☐	NULL		
21	bothFeetMove_ID	INT	11	☐	☑	☐	NULL		
22	bothFeetForward_ID	INT	11	☐	☑	☐	NULL		
23	bothFeetBackward_ID	INT	11	☐	☑	☐	NULL		

Figura E.68: Esquema de la tabla Mindstormconfiguration

E.4.3. Tablas de Evaluaciones

- Evaluation - Pag. 147

Add / Remove / Clear / Up / Down

Name	Type / Length	Algorithm
PRIMARY KEY	PRIMARY	
ID	UNIQUE	
user_ID	KEY	

Columns: Add Remove Up Down

#	Name	Datatype	Length/Set	Unsign...	Allow NULL	Zerofill	Default	Comment	Collation
1	ID	INT	11	☐	☐	☐	AUTO_INCREMENT		
2	type	INT	11	☐	☐	☐	No default		
3	date	DATETIME			☐		No default		
4	started	TINYINT	1	☐	☐	☐	No default		
5	completed	TINYINT	1	☐	☐	☐	No default		
6	profileSerialized	LONGTEXT			☑		No default		
7	user_ID	INT	11	☐	☑	☐	NULL		

Figura E.69: Esquema de la tabla Evaluation

- EvaluationItem - Pag. 147

Add / Remove / Clear / Up / Down

Name	Type / Length	Algorithm
PRIMARY KEY	PRIMARY	
ID	UNIQUE	
evaluation_ID	KEY	
gesturePlayed_ID	KEY	
gestureRecorded_ID	KEY	

Columns: Add Remove Up Down

#	Name	Datatype	Length/Set	Unsign...	Allow NULL	Zerofill	Default	Comment	Collation
1	ID	INT	11	☐	☐	☐	AUTO_INCREMENT		
2	evaluation_ID	INT	11	☐	☑	☐	NULL		
3	gesturePlayed_ID	INT	11	☐	☑	☐	NULL		
4	gestureRecorded_ID	INT	11	☐	☑	☐	NULL		

Figura E.70: Esquema de la tabla EvaluationItems

- EvaluationItemResult - Pag. 148

Add / Remove / Clear / Up / Down

Name	Type / Length	Algorithm
PRIMARY KEY	PRIMARY	
ID	UNIQUE	
evaluationItem_ID	KEY	
conf_ID	KEY	
GestureRecognized_ID	KEY	

Columns: Add Remove Up Down

#	Name	Datatype	Length/Set	Unsign...	Allow NULL	Zerofill	Default	Comment	Collation
1	ID	INT	11	☐	☐	☐	AUTO_INCREMENT		
2	Result	TINYINT	1	☐	☐	☐	No default		
3	evaluationItem_ID	INT	11	☐	☑	☐	NULL		
4	conf_ID	INT	11	☐	☑	☐	NULL		
5	GestureRecognized_ID	INT	11	☐	☑	☐	NULL		

Figura E.71: Esquema de la tabla EvaluationItemResult

- EvaluationItemResultDTW - Pag. 148

Add
Remove
Clear
Up
Down

Name	Type / Length	Algorithm
PRIMARY KEY	PRIMARY	
ID	UNIQUE	
EvaluationItemResult_ID	KEY	
Gesture_ID	KEY	

Columns: Add Remove Up Down

#	Name	Datatype	Length/Set	Unsign...	Allow NULL	Zerofill	Default	Comment	Collation
1	ID	INT	11	☐	☐	☐	AUTO_INCREMENT		
2	distance	DOUBLE		☐	☐	☐	No default		
3	EvaluationItemResult_ID	INT	11	☐	☑	☐	NULL		
4	Gesture_ID	INT	11	☐	☑	☐	NULL		

Figura E.72: Esquema de la tabla EvaluationItemResultsDTW

- EvaluationResult - Pag. 149

Add
Remove
Clear
Up
Down

Name	Type / Length	Algorithm
PRIMARY KEY	PRIMARY	
ID	UNIQUE	
evaluation_ID	KEY	
conf_ID	KEY	

Columns: Add Remove Up Down

#	Name	Datatype	Length/Set	Unsign...	Allow NULL	Zerofill	Default	Comment	Collation
1	ID	INT	11	☐	☐	☐	AUTO_INCREMENT		
2	Success	INT	11	☐	☐	☐	No default		
3	Total	INT	11	☐	☐	☐	No default		
4	evaluation_ID	INT	11	☐	☑	☐	NULL		
5	conf_ID	INT	11	☐	☑	☐	NULL		

Figura E.73: Esquema de la tabla EvaluationResults

E.4.4. Enumerados

- ConfigurationProfileType - Pag. 149
- ConnectionType - Pag. 150
- DistanceTypes - Pag. 150
- EvaluationType - Pag. 150
- GestureType - Pag. 151
- Language - Pag. 151
- RecognitionType - Pag. 151
- RotateType - Pag. 151

Apéndice F

Comandos SQL

En esta sección se mostrarán algunos comandos SQL utilizados para extraer resultados de las evaluaciones:

F.1. Distancia gestos detectados

La sentencia SQL para obtener las distancias de los gestos con menor distancia, suponiendo la evaluación número 4 y el perfil de configuración 121, es:

```
select distance from evaluationitemresultdtws
   eirdtw LEFT JOIN evaluationitemresults eir
   ON eirdtw.EvaluationItemResult_ID = eir.ID
   LEFT JOIN evaluationitems ei ON
   eir.evaluationItem_ID = ei.ID WHERE
   ei.evaluation_ID = '4' AND eir.conf_ID =
   121 AND distance in (select min(distance)
   from evaluationitemresultdtws where
   eir.ID=EvaluationItemResult_ID);
```

F.2. Distancia del segundo gesto más cercano

La sentencia SQL para obtener las distancias de los gestos con segunda menor distancia, suponiendo la evaluación número 4 y el perfil de configuración 121, es:

```
select distance from evaluationitemresultdtws
   eirdtw LEFT JOIN evaluationitemresults eir
   ON eirdtw.EvaluationItemResult_ID = eir.ID
   LEFT JOIN evaluationitems ei ON
   eir.evaluationItem_ID = ei.ID WHERE
   ei.evaluation_ID = '4' AND eir.conf_ID =
   121 AND distance in (select min(distance)
   from evaluationitemresultdtws where
   eir.ID=EvaluationItemResult_ID AND distance
   NOT IN (select min(distance) from
   evaluationitemresultdtws where
   eir.ID=EvaluationItemResult_ID ))
```

Bibliografía

[1] Gonzalo Bailador, Daniel Roggen, y Gerhard Tröster. Real time gesture recognition using continuous time recurrent neural networks. Inf. téc., Fac. Informática (UPM), Wearable Computing Lab (ETH Zürich), Wearable Computing Lab (ETH Zürich), 2007.

[2] Ricard Bellman y Robert Kalaba. On adaptative control processes. *Automatic Control, IRE Transactions*, 4:1–9, 1959.

[3] Thomas M. Cover y Joy A. Thomas. *Elements of Information Theory.* Wiley-Interscience, New York, NY, USA, 1991. ISBN 0-471-06259-6.

[4] Olivier Faugeras, Bernard Hotz, Hervé Mathieu, Thierry Viéville, Zhengyou Zhang, Pascal Fua, Eric Théron, y Projet Robotvis. Real time correlation-based stereo: Algorithm, implementations and applications. Inf. téc., INRIA, 1996.

[5] Pengyu Hong, Matthew Turk, y Thomas S. Huang. Constructing finite state machines for fast gesture recognition. *15th International Conference on Pattern Recognition (ICPR'00)*, 4:691–694, 2000.

[6] Pengyu Hong, Matthew Turk, y Thomas S. Huang. Gesture modeling and recognition using finite state machines. *In Proceedings of the Fourth IEEE International Conference and Gesture Recognition*, págs. 410–415, 2000.

[7] Meinard Müller. *Information Retrieval for Music and Motion.* Springer, 2007.

[8] Kouichi Murakami y Hitomi Taguchi. Gesture recognition using recurrent neural networks. *CHI '91 Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, págs. 237–242, 1991.

[9] Ricardo Ramos. Transformaciones lineales en 3d. Inf. téc., Universidad de Oviedo, 2010.

[10] Tevfik Metin Sezgin y Randall Davis. Hmm-based efficient sketch recognition. *Proceedings of the 10th international conference on Intelligent user interfaces*, págs. 281–283, 2005.

[11] J. Shotton, A. Fitzgibbon, M. Cook, T. Sharp, M. Finocchio, R. Moore, A. Kipman, y A. Blake. Real-time human pose recognition in parts from single depth images. En *Proceedings of the 2011 IEEE Conference on Computer Vision and Pattern Recognition*, CVPR '11, págs. 1297–1304. IEEE Computer Society, 2011. ISBN 978-1-4577-0394-2.

[12] Jamie Shotton, Andrew Fitzgibbon, Toby Sharp Mat Cook, Mark Finocchio, Richard Moore, Alex Kipman, y Andrew Blake. Real-time human pose recognition in parts from single depth images. Inf. téc., Microsoft Research Cambridge - Xbox Incubation, 2011.

[13] Jianfeng Wang, Cha Zhang, Wenwu Zhu, Zhengyou Zhang, Zixiang Xiong, y Philip A. Chou. 3d scene reconstruction by multiple structured-light based commodity depth cameras. En *ICASSP*, págs. 5429–5432. IEEE, 2012. ISBN 978-1-4673-0046-9.